Dicembre 1984/Gennaio 1985 - Lire 3.000 - Anno II - N° 6 - Sped. Abb. Post. GR III/70 (CR) - Distr. MePe

COMMODORE

MENSILE PER UTENTI DI VIC 20 - C64 - C16 - PLUS-4

COMMODORE

DIRETTORE
Gloriano Rossi

REDAZIONE/COLLABORATORI
Eugenio Coppari, Giancarlo De Cobelli, Marco De Martino, Marco De Rosa, Valerio Ferri, Francesco Gatti, Mariangela Guardione, Giulio Marcozzi, Mauro Massetti, Ernesto Sidoti, Renzo Zonin.

SEGRETERIA DI REDAZIONE
Maura Ceccaroli, Piera Perin

GRAFICA e IMPAGINAZIONE
Francesco Amatori, Renato Caruso

EDIZIONI
Systems Editoriale S.r.l.

DIFFUSIONE e ABBONAMENTI
Marina Vantini

DIREZIONE, REDAZIONE
Viale Famagosta, 75 - 20142 Milano
Tel. 02/8467348 - Autorizzazione del Tribunale di Milano N. 103 del 25/2/84

DIRETTORE RESPONSABILE
Agostina Ronchetti

PUBBLICITA'
• Milano: Mirco Croce (coordinatore), Michela Prandini, Giorgio Ruffoni, Claudio Tidone, Villa Claudio
Segretaria: Liliana De Giorgio
V.le Famagosta, 75 - 20142 Milano
Tel. 02/8467348
• Roma: Spazionuovo
Via P. Foscari, 70 - Tel. 06/8109679
• Torino: Spazioblu
Via Filadelfia, 50 - 10134 Torino
Tel. 011/327617

STAMPA
Litografica - Busto Arsizio

Concessionario esclusivo per la diffusione MEPE Spa Via G. Carcano, 32 Milano

Spedizione in abbonamento postale Gruppo III/70

Prezzo della rivista L. 3.000
Numero arretrato L. 6.000
Abbonamento annuo L. 28.000
I versamenti vanni indirizzati a: Commodore C.C. V.le Famagosta, 75 20142 Milano, mediante assegno bancario, utilizzando il c/c postale N. 31532203

Per i cambi di indirizzo, indicare, oltre naturalmente il nuovo, anche l'indirizzo precedente, ed allegare alla comunicazione l'importo di L. 500 anche in francobolli.

LA POSTA

• Essendo possessore di un Commodore 64 ho avuto modo di lavorare con la grafica di questo computer. Possedendo un monitor, tutte le belle figure che creavo col 64 erano nitide e perfette; il problema è sorto quando ho registrato su videoregistratore dei programmi grafici. Allora se collegavo il VCR al monitor si vedeva bene, ma quando ho portato la video cassetta ad una TV locale per trasmetterla via etere, se non andava in onda tutto OK, quando veniva trasmessa appariva una striscia nera in basso che ballava rovinando il tutto. Allora chiedo come posso risolvere questo mio problema?

(Antonio della Colletta)

□ *La causa può essere individuata in una rivelazione di radiofrequenza. Ciò può essere locato sulla testina rotante o sul circuito di amplificazione. Questa interferenza può causare un tentativo di sgancio dei sincronismi. Provi a verificare le massa.*

• Sono in possesso del Commodore 64 e vorrei sapere se è possibile tenere contabilità/IVA/fatturazione, a livello artigianale o di piccole aziende. Chiaramente questo lavoro verrebbe effettuato per conto terzi.

(Vecchi Emilio)

□ *Si e no. Ma che risposta è questa? In realtà non si può dare decisamente una risposta affermativa o negativa. Tutto dipende dalle necessità del momento o meglio dal cliente o da alcune altre caratteristiche. Esistono in commercio alcuni package di contabilità, fatturazione e magazzino (più o meno tutti copiati dalla vecchia serie 3000 della Commodore) che hanno la particolarità di "girare" e più non commento. Non confondiamo il C64 con altri tipi di computer fatti appositamente per applicazione professionale/gestionale. Il Commodore 64 è una macchina magnifica, ma cerchiamo di conoscerne i limiti e senza dubbio potremo apprezzarla ancora di più.*

• Recentemente ho acquistato un Commodore 64 per mia figlia e siamo alle prime armi nell'apprendere il BASIC. Nel copiare dei listati incontriamo spesso dei simboli grafici, linee o parentesi quadre ecc. che non riusciamo a capire e trasferire sul video, in particolare i simboli in neretto. Probabilmente non usiamo correttamente qualche tasto o funzione per cui vorremmo un VS. consiglio per superare l'ostacolo. Forse esiste anche qualche piccolo trucco dovuto all'esperienza che non conosciamo. Ringraziamo per la cortese risposta, se possibile direttamente, per evitare l'attesa per noi molto lunga, di vederla pubblicata sulla rivista.

(Zucconi Giorgio)

□ *Proprio per questa ragione su Commodore riportiamo sempre (tranne qualche rara eccezione) i listati in un formato particolare. Cioè ogni comando di controllo viene interpretato e posto tra parentesi quadre. Ogni qualvolta si incontrano queste parentesi si deve eseguire la digitazione del comando, così ad esempio [up] vuole dire che deve essere premuto il tasto di controllo cursore (crsr) con le freccette verticali. Non spaventatevi quindi, con l'esperienza questi problemi non sussisteranno più.*

• Possiedo un Commodore 64 e nei ritagli di tempo cerco di fare qualche programmino di matematica, applicazioni di fisica, ecc. In tali programmi, il più delle volte, è necessario inserire una funzione durante l'esecuzione; in questi casi ricorro al LIST-linea definizione della funzione, fermando il programma e dando il RUN per ricominciare. C'è la possibilità con qualche accorgimento o qualche routine in linguaggio macchina, d'inserire una tramite un input normale?

(Stefano Masini)

□ *Carissimo Stefano Masini, il problema osservato nelle sue righe è indubbiamente molto interessante. Mi ricordo, circa 8 anni fa agli albori dei computer Commodore, che tale problema è stato risolto con questo semplice artifizio:*

- *si prevede sul listato una riga completamente riempita di ":" (due punti);*
- *si individua la posizione in memoria di detta riga;*
- *con opportune POKE si costruisce la formula necessaria.*

Tutto ciò è forse più semplice da dire che non da fare, comunque auguri!

• Ho letto con vivo interesse l'articolo CIA (6526) e di conseguenza vorrei sottoporvi il mio caso, che a quanto sò non è solo mio. Mi si è bruciato il 6526 usando il floppy. Come si vede dallo schema il 6526 è connesso alla porta seriale tramite degli inerenti (7406). A questo punto ho portato il Commodore 64 dal mio rivenditore il quale mi ha detto che l'avrebbe spedito alla casa importatrice e che sarebbe tor-

nato dopo 4 lunghi mesi (cose inaudite). Siccome il C64 a me serve per lavoro ho cercato di darmi da fare. La casa importatrice mi ha detto di rivolgermi al distributore di zona. A sua volta questi mi ha informmato che la Commodore non fornisce pezzi di ricambio. mma se avessi inviato a loro il computer forse me lo avrebbero riparato usando pezzi di altri computer rotti. Credo che la Commodore generale sia ubicata in Inghilterra. Così un po' demoralizzato ho cominciato a telefonare a tutti i negozi, inclusi i rivenditori all'ingrosso di componenti elettronici (di Padova, Milano), i quali mi hanno detto che tale integrato è molto richiesto, ma non ce l'hanno. Concludendo (a parte la bolletta SIP) questo integrato l'ho trovato grazie alle conoscenze di un grosso distributore. Capisco che possano essere utili le raccomandazioni per trovare un posto di lavoro, ma non supponevo che ce ne volessero tante per trovare il 6526.
P.S. Lo stesso importatore a me aveva detto di non avere questo integrato.

A questo punto vi chiedo, la prossima volta che avrò bisogno di un integrato a chi devo rivolgermi senza raccomandazioni?

(IK3 ABY Stefano)

☐ *Caro IK3 ABY Stefano, come l'autore dell'articolo anche io sono radioamatore (I2KH) ed ecco perchè, come nostro uso, ti rispondo in tono confidenziale. Ho riportato per intero la tua lettera in quanto ritengo che il problema da te riscontrato sia attuale per molti altri possessori di C64. Ti confermo tutto ciò perchè io stesso per altre ragioni ho avuto dei problemi al quanto similari. Alcune volto ho dovuto portare al centro servizio del rivenditore regionale dei C64 e... dopo lunghe attese, mesi come accusi tu, ho scelto l'opportunità di rivolgermi ad un importatore... privato. A questo punto nacquero altri problemi che riguardano i pezzi delle parti di ricambio. Mi sono dato da fare e, a discapito del prezzo, mi sono assicurato una scorta sufficiente per le mie necessità. Con questo ti ho raccontato la mia Odissea vista in termini più grossi dei tuoi, ma altrettanto importante.*

Perchè la Commodore International non fornisce questi benedetti pezzi di ricambio? Come dare una risposta?

Si possono fare solamente delle ipotesi. Innanzitutto la Commodore è una multinazionale con sede in California, ed è direttamente produttrice degli integrati di cui sopra (MOS Tecnology); quindi detta produzione è fatta per sè. Questo fatto può indurre solo ad ipotesi del tipo: a/ la produzione è appena sufficiente per la costruzione dei computer; oppure b/ per qualche strategia di mercato l'approvigionamento delle parti di ricambio è limitato esclusivamente alle più strette necessità. Per noi la cosa non va bene. Ma possiamo forse dare tutti i torti alla Commodore se, per strategia industriale, avesse deciso di agire in questo modo?

• **Sono un dipendente di una media azienda del centro Italia e nell'ambito della ditta occupo un posto di programmatore COBOL su un DPS4 della Honeywell. Ho acquistato il Commodore 64 con il suo floppy e la stampante. Mi sto divertendo con il BASIC, ma mi sorge un dubbio: si può avere il COBOL sul Commmodore 64?**

(Salvatore Napolitano)

☐ *Non si può avere tutto dalla vita! Le prestazioni del tuo Commodore sembrerebbero limitate al BASIC e l'Assembler. Perchè al condizionale? Infatti non è completamente vera questa limitazione. Sul Commodore 64 possono girare altri linguaggi di programmazione come ad esempio il Lisp, il COMAL, il Pascal ed altri, il tutto semplicemente "caricando" il relativo interprete/compilatore. Per arrivare invece al COBOL si sa che le mega case di software non hanno mai intrappreso l'azione di trasportare questo linguaggio sul microprocessore 6502 o 6510.*

Cosa si può fare allora?

La Commodore ha pensato di realizzare un optional che guarda cosa può aiutare a risolvere questo problema: la scheda CPM.

Inche cosa consiste questo optional? Prima di rispondere a questa domanda occorre sapere che il CPM è un sistema operativo della Digital Research sviluppato per il microprocessore Z80 ed in seguito implementato sull'8088. Sotto questo sistema operativo girano molti linguaggi tra i quali anche un tipo di COBOL. La scheda CPM è costituita da una parte elettronica che comprende il famoso Z80 da connettere al 64 (sposalizio tra Z80 e 6510) ed un dischetto formato 1541 che contiene il sistea operativo CP vero e proprio. A questo punto procurarsi un qualsiasi linguaggio previsto dal CPM non dovrebbe essere cosa difficile e... quindi: si, sul Commodore 64 c'è anche il COBOL.

COMMODORE ONE

Durante il Salone della Nautica di Genova il ketch di 18 metri che porta il nome di Commodore One, e rimasto ormeggiato nella darsena della mostra e in quell'occasione Giorgio Moser ha illustrato alla stampa presente il suo prossimo viaggio verso gli Stati Uniti, dove realizzerà una serie televisiva dal titolo "Ventimiglia leghe dentro l'America".

Il viaggio/documentario prenderà le mosse da San Salvador, dove si concluse quello di Cristoforo Colombo, e, una volta attraversato il Golfo del Messico, Commodore One risalirà il Mississipi per 4000 chilometri fino a St. Louis. Da lì attraverso l'Illinois entrerà nel lago Michigan, poi nei laghi Huron, Erie e Ontario. Scendendo quindi lungo l'Hudson fino a New York, Commodore One imboccherà l'Intercoast Channel, che corre parallelo all'Atlantico, per sbucare nel Golfo del Messico davanti alla Florida.

Un round trip di 20.000 leghe nel cuore dell'America di esploratori e pionieri, lungo le "water-way". Un itinerario inedito che, da Colombo ad oggi, nessuno ha mai verificato, che la macchina da presa ha inspiegabilmente ignorato.

Commodore One: un gioiello dell'elettronica

Come può entrare un computer Commodore su una barca?
Il calcolatore è in grado di svolgere un'infinità di funzioni, dal check-up permanente del motore e dell'impianto elettrico alla gestione delle trasmissioni via radio con la terraferma, al collegamento con le apparecchiature di bordo. Il tutto con computer Commodore 64 Executive.

Su Commodore One, il computer è protagonista soprattutto dei collegamenti in ricetrasmissione. Oltre al radiotelegrafo, un Commodore 64 può gestire la trasmissione automatica del messaggio di soccorso con denominazione e posizione della barca. Per il calcolo della rotta e delle informazioni fondamentali in navigazione, Commodore One è collegata con il sistema satellitare, con il Meteosat e con il sistema Loran. Nella navigazione sottocosta, sullo schermo del computer vengono visualizzate le carte nautiche relative al tratto di costa interessato, e viene tracciata in ogni istante la posizione della barca e la rotta seguita.

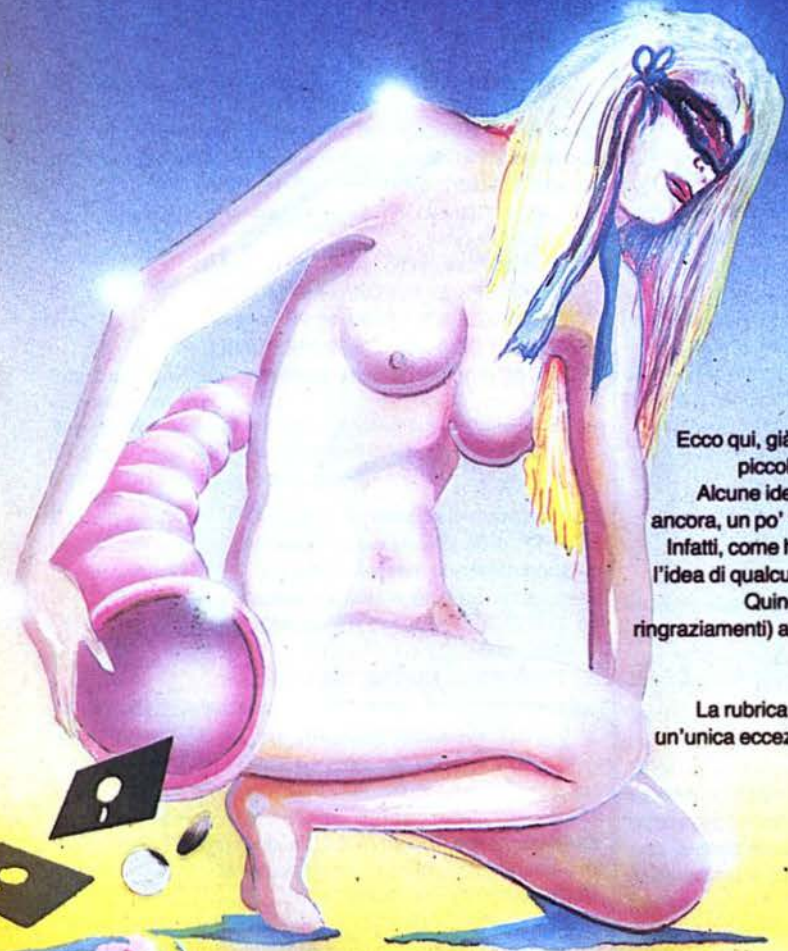
Il controllo computerizzato della strumentazione di bordo (bussola elettronica, timone a vento, pilota automatico) riduce al minimo le possibilità di errore. Oltre a fare da ufficiale di rotta e da

skipper, il computer si trasforma di volta in volta in meccanico, medico e cambusiere con l'aiuto di programmi studiati appositamente per la spedizione. Uno degli Executive fornisce sul video le informazioni necessarie per l'assistenza tecnica in caso di avaria. E, per finire, un programma sviluppato da un'équipe di medici del CNR trasforma il computer in un medico elettronico in grado di prestare le prime cure in caso di malore, mentre un ricettario elettronico assicura una rotazione dei menù settimanali.

Quali sono le possibili applicazioni richiamabili dai C 64 presenti su Commodore One? Vediamone il "Menù":

- Check-up del motore
- Check-up del generatore
- Check-up dell'impianto elettrico
- Trasmissione del messaggio di soccorso S.O.S. con denominazione e posizione della barca in telegrafia e in fonia con il sintetizzatore vocale
- Controllo della presenza dell'acqua in sentina con intervento automatico delle pompe
- Controllo livello nafta e consumo del motore
- Controllo dell'ottimizzazione delle vele
- Interfaccia con il satellite, con il loran, con il radar ed il radiogoniometro
- Interfaccia con il pilota automatico
- Interfaccia con la radio VHF e HF con emissioni in FM, SSB, RTTY e Telegrafia
- Calcolo del punto nave con l'uso del sestante
- Manuale d'uso degli strumenti di bordo
- Il meccanico di bordo
- Il medico di bordo
- Il cuoco di bordo
- Controllo del timone a vento
- Interfaccia con centralina elettronica metereologica, con strumenti del vento, con il log e la bussola elettronica
- Stesura del giornale di bordo
- Magazzino attrezzature
- Indicazione della forza del vento, del mare e delle maree
- Videotel
- Collegamento diretto con la terraferma a una banca dati
- Interfaccia con l'ecoscandaglio
- Contabilità per la barca
- Giochi
- Portolano
- Programmi per la realizzazione delle vele
- Programmi per la realizzazione dello scafo
- Ricezione immagini tramite facsimile
- Manutenzione programmata della barca
- Antifurto
- Controllo rotazione della barca all'ancora e relativa lunghezza della catena in relazione al fondale
- Segreteria telefonica computerizzata
- Trasmissione programmata di messaggi
- Controllo incendi a bordo con attivazione estintori

Chi è Giorgio Moser

Regista, laureato in giurisprudenza, giornalista professionista, critico cinematografico, inviato speciale. Aiuto regista di Germi, Blasetti e Edgar Ulmer.
Ha realizzato più di 30 documentari in Africa e in Asia, fra cui i più significativi sono: "Il pescatore e la ballerina", "È nata una stella", "Sul fiume dei leopardi", e "Gli zingari del mare".
Ha diretto i films quali ad esempio "Continente perduto", "Un po' di cielo", "Calypso, violenza segreta" e "Un reietto delle isole", premiato a Sorrento.
Ha diretto diverse serie televisive; tra le altre: "Vado a vedere il mondo, capisco tutto e torno", "Sulla rotta di Magellano", "Le montagne della luce", "Le dieci incarnazioni di Vishnù" e "Avventure africane di un bersagliere ciclista".
Ha pubblicato i seguenti libri: "Alla scoperta di Magellano" (Fabbri), "I pascoli del sole" (Mursia), "Il viaggio di Magellano" (Mursia), "Passaggio a Sud-Est" (ERI) e "Le montagne della luce" (ERI).

Trent'anni di viaggi quasi ininterrotti fra Africa, Asia, America e arcipelaghi dei due grandi Oceani lo hanno indotto a costruirsi una barca-laboratorio (due alberi, 18 metri), perfettamente attrezzata per le riprese.

Cornucopia

a cura di **Gloriano Rossi**

Ma quando arrivano i $NN dei lettori?
Ecco qui, già da questa terza puntata di Cornucopia una piccola raccolta di $NN tutta redatta da Voi lettori.
Alcune idee sono sono ottime, altre un po' meno, altre ancora, un po' "scarsine". Ma non diamo giudizi avventati.
Infatti, come ho già avuto modo di dire: tutto può servire e l'idea di qualcuno può andare altrettanto bene per un altro.
Quindi voglio fare i miei complimenti (uniti ai miei ringraziamenti) a quei numerosi lettori che, penna alla mano, hanno inviato i propri elaborati.
Cosa devo dire d'altro? Continuate così!
La rubrica Cornucopia è Vostra in tutto e per tutto, con un'unica eccezione: il mio controllo di validità ed esattezza delle notizie.
Continuate ad inviare i vostri $NN a:

Spett. Rivista COMMODORE
rubrica Cornucopia
Gloriano Rossi
Viale Famagosta 75
20142 Milano

$1C

Espansione o no! Per far partire un programma non espanso con l'espansione inserita nel VIC 20 bisogna battere:

POKE 43,0
POKE 44,32

e controllare con la PEEK

PEEK (43)
PEEK (44)

L'indirizzo di partenza di un programma Basic che utilizza una espansione superiore ai 3K è 8192 e si ottiene calcolando il valore della locazione 44 * 256 + il contenuto della locazione 43.

(Luciano Rossi I2EAY)

$1D

Routine per interrogazione joystick. Ecco qui una breve routines utile per sapere in che posizione è stato spostato il joystick.

```
100 POKE DD,127:P=PEEK (P2) AND 128
110  J0=-(P=0)
120  POKE DD,255:P=PEEK (P1)
130  J1=-(P AND 8)=0)
140  J2=-(P AND 16)=0)
150  J3=-(P AND 4)=0)
160  IF J0=1 THEN X=X+1
170  IF J1=1 THEN Y=Y+1
180  IF J2=1 THEN X=X-1
190  IF J3=1 THEN Y=Y-1
200  IF Y > 112 THEN Y=112
210  IF X > 136 THEN X=136
```

(Luciano Rossi I2EAY)

$1E

PRINT AT. (n1). La maggior parte dell'utenza Commodore dimostra non gradire l'uso eccessivo dei tasti per il posizionamento del cursore poichè, creando degli strani simboli, non rende agevole la lettura e la comprensione di un listato di programma.

A tale scopo si può facilmente ricorrere all'uso della routine "PLOT" del kernal. Il suo utilizzo permette di simulare efficacemente la nota istruzione BASIC PRINT AT (ri,co) oltre che una nuova versione della INPUT, una INPUT AT (ri,co)

Il seguente listati mostra il suo uso.

```
10 RI=10:CO=15:GOSUB 500
20 PRINT "COMMODORE"
..........................................................
..........................................................
500 REM ******************************************
510 REM   *    SUBRUTINE    *
520 REM   *POSIZIONAMENTO   *
530 REM   *   CURSORE   *
540 REM ******************************************
550 POKE 781,RI:POKE 782,CO
560 SYS 65520
570 RETURN
```

In questo modo, la scritta Commodore verrà visualizzata a riga 10 e colonna 15 dello schermo.

(Franco Silvestri & Dario Renelli)

$1F

Load Error. Se dopo aver caricato un programma da cassetta compare il LOAD ERROR c'è ancora qualche speranza di salvare qualche cosa.

Provare a dare il LIST, se il programma non presenta segni strani allora è fatta!

Non date RUN ma battete in modo diretto:

```
POKE45,PEEK(831): POKE47,PEEK(831)
POKE49,PEEK(831): POKE46,PEEK(832)
POKE48,PEEK(832): POKE50,PEEK(832)
```

Ora il programma è salvo e potete registrarlo nuovamente.

(ANtonio Adorno)

$20

Simulazione comandi da tastiera C64. In pratica sfrutta il BUFFER della tastiera dove si possono deporre fino a dieci caratteri (dovendo mettere anche il RETURN per mandare in esecuzione il comando avrà al massimo nove caratteri):

```
10  L=LEN(C$): FORI=1TOL:POKE630+I,
    ASC(MID$ (C$,1,1)):NEXT
20  POKE630+I,13:POKE198,1+1
```

Il comando dovrà essere inserito nella variabile C$.

(Antonio Adorno)

$21

Tasti funzione (VIC 20 e C64). Se dopo un GET di attesa per un comando da tastiera come ad esempio:

```
10 GET A$:IF A$=""THEN 10
```

inserite una GOSUB 100, il programma darà in uscita dalla subroutine il numero corrispondente al tasto funzione premuto sulla variabile F; se si preme un qualsiasi altro tasto si ha invece il valore 0 (esclusi i tasti CTRL, RESTORE, SHIFT e RUN STOP). Per vedere come funziona aggiungere:

```
20 PRINT F:GOTO 10
```

La subroutine è la seguente:

```
100 K=PEEK(197): SH=PEEK(653(AND1
110 F=-(K=4)-3* (K=5) -5* (K=6)-7* (K=3)
120 IF F>0 THEN F=F+SH
130 RETURN
```

(Antonio Adorno)

$22

Trovare l'errore nel DATA. Di solito nei programmi con molti data succede di sbagliare la copiatura; se si scrive un DATA maggiore di 256; si ottiene un ILLEGAL QUANTITY ERROR; che solitamente ci fa impazzire per trovare quello errato. Il nostro CBM-64 può però aiutarci nella ricerca in quanto sa dove ha trovato l'errore. Per cui dopo aver avuto la segnalazione di errore digitare in modo diretto:
PRINT PEEK (63)+PEEK (64)*256
ed otterremo il numero di riga in cui si trova il DATA errato.

(Antonio Adorno)

$23

AUTORUN per disco. Un solo insieme di comandi ci permette di scrivere RUN+RETURN dopo il LOAD, ci basta infatti:
LOAD"nome prg.",8:SHIFT+RUN STOP

(Antonio Adorno)

$24

Merge per C64. Il tutto è molto facile: per fare il MERGE (la fusione) di due programmi si opera così:
avuto in memoria il primo programma battere:
POKE43,PEEK (45)-2:POKE44,PEEK(46)
caricare normalmente il secondo programma e quindi rimettere tutto a posto battendo:
POKE43,1:POKE44,8

(Antonio Adorno)

$25

PRINT AT con SYS (n.2). Volendo simulare in modo chiaro l'istruzione PRINT AT, sul C64 si può ricorrere a:
100 POKE211,X: POKE214,Y: SYS58732:
PRINT "quello che vuoi" X e Y sono ovviamente i valori di riga e colonna. Non devono superare rispettivamente 40 e 24.

(Marco Melloni)

$26

Oscuramento. Per rendere più veloci i programmi in BASIC, specialmente durante la lettura di massicce quantità di DATA si può mandare in "BLANK" il video. Per fare ciò digitare:
POKE53265,PEEK (53265) AND239
Per riattivare lo schermo digitare invece:
POKE53265,PEEK (53265) OR16

(Marco Melloni)

$27

Modifichiamo le parole BASIC. Se a qualcuno non dovesse piacere il BASIC dell'interprete del C64 può sempre provare con:
FORJ = 40960TO49151: POKEJ,PEEK (J): NEXTJ POKE1, 54
Così facendo il BASIC è stato ricopiato dalla ROM nella RAM sottostante. Ora si provi per esempio:
POKE41489,66
Il computer dovrebbe scrivere BEADY invece di READY.
Per ripristinare le condizioni originali basterà digitare:
POKE1,55

(Marco Melloni)

$28

Altre insidie del BASIC (il manuale ne parla ma...). A volte nel redigere programmi si usa assegnare alle variabili dei nomi, anzichè lettere senza significato, per riconoscerle facilmente: bisogna stare attenti a non inserire involontariamente parole BASIC ad esempio:
QUESTO=3: PRINT QUESTO
SIFONE=4: PRINT SIFONE
PRENOTA=5: PRINT PRENOTA
TENDA=6: PRINT TENDA
TO, IF, NOT, END negli esempi danno il SINTAX ERROR.

(Riccardo Iuliani)

$29

I limiti dell'elaboratore. Se provate ad inserire comandi come:
10 FORT=1TO25: PRINT T 2: NEXT
oppure
10 INPUT A
20 IF SQR(A)=INT (SQR(A)) THEN PRINT
SQR(A): END
30 PRINT A
otterrete risultati a prima vista sconcertanti. Nel primo caso invece dei numeri interi ne verranno mostrati alcuni decimali. Nel secondo caso il vostro computer, se inserite un quadrato perfetto come 9, stamperà quest'ultima cifra eseguendo quindi a torto la linea 30, invece di fermarsi alla linea 20 dopo aver stampato 3.

E' chiaro che non vi sono errori di sorta, e che le linee BASIC proposte sono formalmente corrette. L'arcano è spiegato dalle limitazioni proprie dell'elaboratore (in questo caso del BASIC residente) le cui approssimazioni portano errori di questo tipo.

Nel primo caso è sufficiente scrivere:
10 FORT=1 TO25:PRINT INT(T 2): NEXT

Nel secondo caso potremo aggirare l'ostacolo, appesantendo però un po' le cose, usando la funzione stringa STR$:
10 INPUT A:R=VAL (STR$(SQR(A)))

20 IFR=INT(R) THEN PRINT R:END
30 PRINT A

Tutto funzionerà correttamente.

(Dino Ticli)

$2A

Scrool su VIC 20. E' simpatico dar vita ad un proprio programma facendo scomparire e ricomparire il video lentamente e gradatamente insieme a tutto ciò che vi compare. Il mio programma offre questa possibilità Ecco il listato:

```
10 REM ******************************************
15 REM  *  video verso il basso  *
20 REM ******************************************
30 PRINT "[CLR]":A=39
40 POKE 36881,A:A=A+1
50 FOR T=1 TO 30:NEXT T
60 IF A=153 THEN GOTO 80
70 GOTO 40
80 REM ******************************************
85 REM  * video verso l'alto *
90 REM ******************************************
100 POKE36881,A:A=A-1
110 FOR T=1 TO 30:NEXT T
120 IF A=37: THEN END
130 GOTO 100
```

Questo programma gira sul VIC 20 ma può essere adattato anche sul C64: si può inserire in qualsiasi altro programma come maschera iniziale o finale e chi più ne ha più ne metta. Si esce dal programma tramite la riga 120, cioè quando il video è tornato alla condizione normale, si regola la velocità di discesa e di salita rispettivamente nelle righe 50 e 110.

(Marco Olimperi)

$2B

Lampeggio. Ecco come fare far lampeggiare una scritta (non più di 39 caratteri) sempre al centro dello schermo.

```
10 N$="PIETRO"
20 PRINT SPC(200+(39-LEN(N$))/2)N$
30 FOR X=0 TO 500:NEXT:PRINT"[CLR]"
40 FOR Y=0 TO 500:NEXT:GOTO 10
```

La velocità di lampeggiamento dipende dai cicli FOR/NEXT nelle righe 30 e 40. Naturalmente possono essere aggiunti per il colore della scritta e/o cambiarne la posizione definita in 20.

(Pietro Marini)

$2C

La riga fantasma. Già presentata nel $0C è più facilmente ottenibile con l'uso dei ":".

```
100 :
```

Si ottiene così, un segno spaziatore dei vari settori di programma.

(Andrea Bartolini)

$2D

PRINT AT senza SYS (n.3). Si può ottenere una sorta del comando PRINT AT usando la locazione 214. La linea di programma che segue permette di stampare una stringa a partire dalla posizione riga-colonna (X,Y) desiderata:

N.B: andando a stampare, nell'ultima riga interviene lo scrool automatico verso l'alto dell'intera pagina.

```
10 POKE214,Y:PRINT"":PRINTTAB(X)"
   [up]" "messaggio"
```

Dove:

Y=riga (0-24) (24 aziona lo scrool)
x=colonna (0-39)

Esempi:

```
10 PRINT "[CLR]":REM PULISCE LO SCHERMO
20 POKE214,0:PRINT"":PRINTTAB(0)"[up]""CIAO"
```

Questo programma stampa ad Home (0,0) la scritta CIAO.

(Andrea Bortolini)

$2E

Trucchetti per il VIC 20. Ecco alcuni interessanti effetti ottenuti con le seguenti locazioni di memoria del mio VIC 20.

POKE 772,0: Lista di un programma solo i numeri di riga.

POKE 788,220: Elimina il lampeggio del cursore. Utile nella stesura dei programmi per chi fosse disturbato dal pulsante. Unica avvertenza ripristinare la locazione originaria prima di mandare in esecuzione il programma.

POKE 788,220: Come sopra però lascia il segno del cursore ad ogni RETURN. Potrebbe essere utile per delimitare il fine riga.

POKE 788,235: Elimina il cursore.

Il ripristino delle locazioni originarie si ottiene naturalmente premendo contemporaneamente i tasti RUN/STOP e RESTORE.

(Giacomo Calabrese)

$2F

FOR... NEXT. In una serie di cicli FOR... NEXT concatenati invece di ripetere il NEXT si può operare nel seguente modo:

Metodo tradizionale

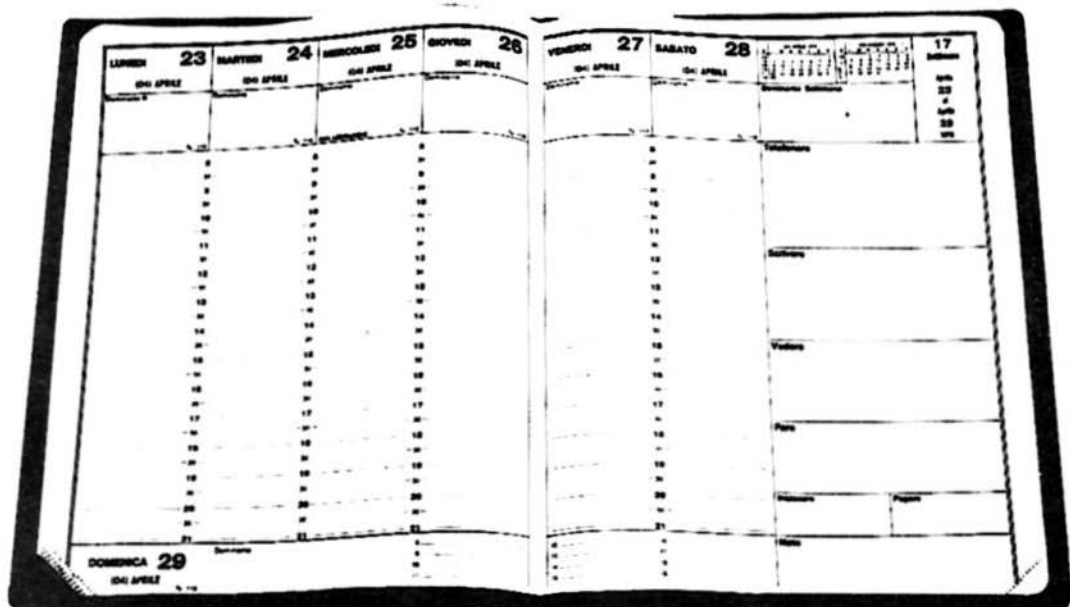
```
10 FOR I=1 TO 10:FOR B=1 TO 20
20 NEXT:NEXT
```

Alternativa alla riga 20

```
20 NEXT B,A
```

(Giulio Spinelli)

L'AGENDA PER GLI APPUNTAMENTI

di **Francesco Gatti**

La memoria è sempre stato un problema per tutti, chi più chi meno. Neanch'io faccio eccezione, tanto è vero che sono un maestro nel far nodi ai fazzoletti, uno dei miei hobby preferiti. Altri sistemi che adotto: annotare su pacchetti di sigarette che si accumulano sulla scrivania, foglietti adesivi con i quali tapezzo la mia stanza e la macchina, piastrelle disegnate con pennarelli.
Insomma metodi molto efficaci senza i quali scorderei parecchie cose. I problemi sorgono però quando si tratta di ricordare appuntamenti a lunga scadenza: i foglietti, contenenti informazioni scritte magari così velocemente da sembrare scarabocchi, ritenute già scadute, vengono inesorabilmente buttati tanto che un fatto come il pagamento del bollo della macchina avviene sempre al pomeriggio dell'ultimo giorno consentito, oppure il passaporto necessario per l'estate diventa inutilizzabile per mancanza di marche da bollo ecc. ecc.

Per questo uno dei primi programmi che feci sul COMMODORE C64 fu una specie di DATA BASE, cioè un archivio nel quale annotavo i vari impegni e li memorizzavo di mese in mese.
Data la mia memoria di massa, quella cioè di colore grigiastro, che si perde i bit con una facilità estrema, ogni inizio settimana consulto la tabella del mese, che mi informa in quali giorni ho un impegno. Quindi richiamo i giorni più prossimi, e trovo i messaggi che contengono le varie pagine e... non tiro più 'bidoni'.
Da questo programma ho ottenuto un altro vantaggio, retrospettivo.
Infatti alla domanda 'Dove ero quel giorno?' so rispondere con esatezza evitando così delle 'gaffes'.

Come Funziona

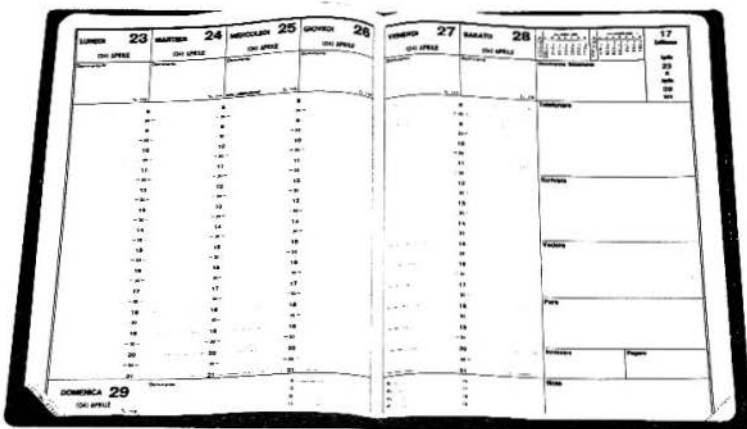
Un programma che deve ricordare i giorni del mese precedente e nello stesso tempo magari i giorni dell'anno successivo, deve contenere per forza un calendario perpetuo, o quasi. Quasi perchè semplificà i calcoli in maniera tale da avere un controllo valido dei giorni fino al 1999: più che sufficiente direi. Il cuore del programma perciò è sito dalla linea **234** alla **322**. Il primo passo per il calcolo da effettuare è accertare se un anno è bisestile o meno. La semplice formula per stabilire ciò è contenuta nella riga **220**: se l'anno è bisestile gli ultimi dei mesi saranno uguali alla stringa **G2$**.
Quindi si passa alla determinazione del nome del primo giorno del mese (righe da **248 a 250**). In questo modo sono solo i nomi dei giorni a ruotare ogni mese, mentre la griglia della data è sempre uguale tranne, per le date dal 29 in avanti. In questi casi infatti occorre vedere quanti giorni in totale ha il mese (variabile GA) e mediante le linee **284-300** e la routine da linea **314** e **322** vengono stampati i giorni successivi al 28.
Per quanto riguarda il resto del programma, si può affermare che nulla è di speciale. Molte ruotines sono state da me già usate per i programmi precedentemente pubblicati. Così ad esempio la procedura di inserimento appunti che non è altro che la titolatrice un pò modificata (Commodore n. 2).
Per lo spostamento verso il basso ho usato la sub-routine di linee **308-310**, alla quale bisogna 'passare' la variabile B con valore uguale a quanti salti verso il basso vogliamo effettuare.

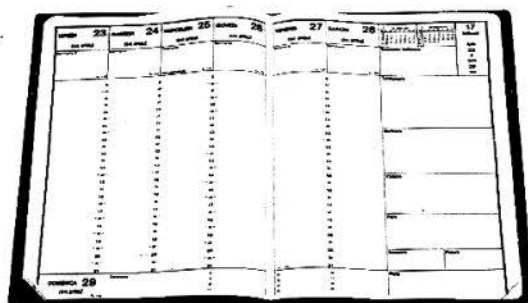
```
C R O S S   R E F E R E N C E

PROGRAMMA : AGENDA APPUN.PRG

-------+----------------- - - -
VAR.   ! LINEA DEL PROGRAMMA
-------+----------------- - - -
       !
A      !   272   274   276   280   362
A$     !   302   564
AN     !   218   220   248   424   544
AN$    !   206   212   218   252   304   392
AP$    !   442   444   446   452   454   460
AP$(   !   128   182   452   454   458   474
       !   498   510   552
AS$    !   552   554
B      !   202   254   268   272   292   308
       !   370   388   422   428   536   542
       !   548
B$     !   308   310
C$(    !   246   286   288   290   296   298
CA     !   250   314   316   318   320   322
D      !   178   182   184   404   406   408
       !   410   412   414   416   418   470
       !   474   478   490   498   500
E      !   480
F      !   538   544   552
F$     !   374   376
FI$    !   304   468   492
FL(    !   128   274   296   402   458   510
G      !   506   510   512
G$     !   394   396
G1$    !   214   222
G2$    !   216   220
GA     !   226   284   286   288   290   398
       !   400
GG     !   398   400   402   404   406   408
       !   410   412   414   416   424   452
       !   454
GG$    !   388   392   396   398   400
GI$    !   308
GIU$   !   130
H      !   490   508   510
I      !   246   458
J      !   350
L      !   550   552   556
M      !   248   250
ME     !   224   226   248   252   398   424
       !   544
ME$    !   210   224   304   392
ME$(   !   154   156   158   160   162   164
       !   166   168   170   200   348   350
       !   386   420   534   540
NG$(   !   134   136   138   140   320   424
       !   544
NM$(   !   128   142   144   146   148   150
       !   152   252   424   544
OG$    !   236   322   418
P      !   270   274   276   292   296   298
       !   312
P$     !   432   434   436   438   440   442
Q      !   448   452   454   460   552   558
       !   560
REV$   !   186   252   274   296
S      !   328   332   516   520
SC$    !   352   354
ST     !   502
T      !   180   182   184   472   474   476
       !   490   494   496   498   500
TT$    !   496   498
V      !   236   294   296   298   300
VI     !   418   424   544
W      !   282   286   288   290   294
X      !   258   354   356   358
XX$    !   220   222   226   398
Y      !   248   250
```

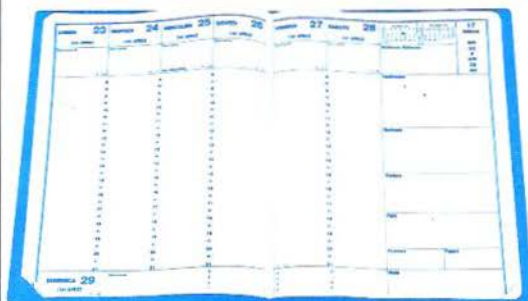
Come si usa

Il programma è composto da un solo menu, dal quale possiamo scegliere le seguenti opzioni:

- Inizializzazione
- Inserimento
- Visualizzazione
- Archiviazione
- Lettura
- Calendario
- Fine

La prima azione da eseguire è quella di servirci dell'opzione Inizializzazione che permetterà di far presente al programma di quale mese volete effettuare le operazioni.
Quindi premendo il tasto 2 si sceglie il giorno nel quale vogliamo memorizzare un determinato messaggio: sarà sufficiente impostare la data desiderata e apparirà sullo schermo una finestra sulla quale possiamo scrivere i nostri appuntamenti o le nostre annotazioni. Quando si è terminato di scrivere il messaggio, premere il tasto funzione 'F7' per poter ritornare al menu principale (infatti il tasto RETURN non viene considerato).
È possibile, tramite l'opzione Calendario, accertarci in quali giorni del mese sono stati inseriti dei commenti. Per questa ragione tutte le date poste in 'reverse' vi indicheranno che lì ci sono delle annotazioni.
Una volta terminata l'operazione di inserimento premete il tasto 4. Dai dati inseriti durante l'Inizializzazione il programma memorizzerà gli appuntamenti col nome del mese prescelto.
Per quanto riguarda la lettura di dati occorre procedere nel seguente modo: prima premete il tasto 1, inserite il mese e l'anno, quindi il tasto 5. Fatto ciò servitevi dell'opzione 3.
Tutto qui! Ed è proprio la semplicità del programma che ne permette l'uso frequente. Parola di utilizzatore.

```
100 REM ***************************
102 REM *  AGENDA APPUNTAMENTI *
104 REM ***************************
106 REM *AUTHOR SOFTWARE          *
108 REM *     GATTI FRANCESCO     *
110 REM ***************************
112 REM *     COMPAT IBILITA'     *
114 REM * VIC 20               NO *
116 REM * VIC 20+EXP           SI *
118 REM * C64                  SI *
120 REM * C16                  SI *
122 REM * PLUS 4               SI *
124 REM * SERIE 4000/8000      SI *
126 REM ***************************
128 DIM NM$(12),AP$(32,2),FL(31)
130 GIU$="[24 DOWN]"
134 NG$(1)="LUNEDI'":NG$(2)="MART
    EDI'"
136 NG$(3)="MERCOLEDI'":NG$(4)="G
    IOVEDI'"
138 NG$(5)="VENERDI'":NG$(6)="SAB
    ATO"
140 NG$(0)="[RVS][GIALLO] DOMENIC
    A [RVOFF][BIANCO]"
142 NM$(1)="GENNAIO":NM$(2)="FEBB
    RAIO"
144 NM$(3)="MARZO":NM$(4)="APRILE
    "
146 NM$(5)="MAGGIO":NM$(6)="GIUGN
    O"
148 NM$(7)="LUGLIO":NM$(8)="AGOST
    O"
150 NM$(9)="SETTEMBRE":NM$(10)="O
    TTOBRE"
152 NM$(11)="NOVEMBRE":NM$(12)="D
    ICEMBRE"
154 ME$(0)="AGENDA APPUNTAMENTI"
156 ME$(1)="[DOWN][RVS] 1 [RVOFF]
         INIZIALIZZAZIONE"
158 ME$(2)="[RVS] 2 [RVOFF]     IN
    SERIMENTO"
160 ME$(3)="[RVS] 3 [RVOFF]     VI
    SUALIZZAZIONE"
162 ME$(4)="[RVS] 4 [RVOFF]     AR
```

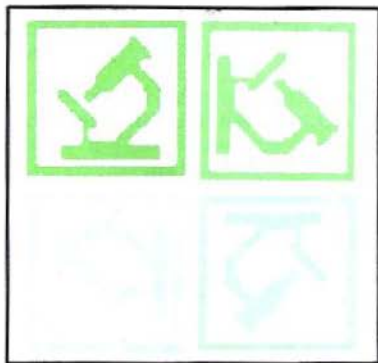
```
    CHIVIAZIONE"
164 ME$(5)="[RVS] 5 [RVOFF]       LE
    TTURA"
166 ME$(6)="[RVS] 6 [RVOFF]       CA
    LENDARIO"
168 ME$(7)="[RVS] 7 [RVOFF]       FI
    NE"
170 ME$(8)="[DOWN]PREMI UN TASTO"
172 REM ***********************
174 REM **  INIZIO PROGRAMMA   **
176 REM ***********************
178 FOR D=0 TO 31
180 FOR T=0 TO 2
182 AP$(D,T)="++"
184 NEXTT,D
186 REV$=CHR$(18)
188 GOTO 346
```

```
190 REM ***********************
192 REM ** INIZIALIZZAZIONE    **
194 REM ***********************
196 POKE 53280,6:POKE 53281,6
198 PRINT"[CLEAR]"
200 PRINT TAB(8)ME$(0)
202 B=4:GOSUB 308
204 PRINT TAB(4)"INSERISCI ANNO:
    AAAA";
206 INPUT "[6 LEFT]";AN$
208 PRINT:PRINT TAB(4)"INSERISCI
    MESE: MM";
210 INPUT "[4 LEFT]";ME$
212 AN$=RIGHT$(AN$,2)
214 G1$="312831303130313130313031
    "
216 G2$="312931303130313130313031
    "
218 AN=VAL(AN$):IF AN<80 THEN 198
220 IF INT(AN/4)=AN/4 THEN XX$=G2
    $:GOTO 224
222 XX$=G1$
224 ME=VAL(ME$):IF ME<1 OR ME>12
    THEN 198
226 GA=VAL(MID$(XX$,ME*2-1,2))
228 REM ***********************
230 REM **     CALENDARIO      **
232 REM ***********************
234 PRINT"[CLEAR]":POKE 53280,2:P
    OKE 53281,2
236 V=0:OG$=""
238 REM ***********************
240 REM **   STAMPA GIORNI     **
242 REM ***********************
244 GOSUB 324
246 FOR I=1 TO 3:C$(I)="   ":NEXTI
248 Y=AN:M=ME-2:IF M<1 THEN M=M+1
    2:Y=AN-1
250 CA=INT(2.6*M-.19)+1+Y+INT(Y/4
    )-34:CA=CA-INT(CA/7)*7
252 PRINT TAB(14);REV$NM$(ME);" "
    AN$
254 B=2:GOSUB 308
256 GOSUB  316
258 FOR X=1 TO 6:GOSUB 314:NEXTX
260 REM ***********************
262 REM *      STAMPA DATA     **
264 REM ***********************
266 PRINT"[HOME]"
268 B=5:GOSUB 308
270 P=16
272 FOR A=1 TO 28:IF A=8 OR A=15
    OR A=22 THEN GOSUB 312:PRINT"
    [HOME]":B=5:GOSUB 308
274 IF FL(A)=1 THEN PRINT TAB(P)R
    EV$A:GOTO 278
276 PRINT TAB(P)A
278 PRINT
280 NEXTA
282 W=0
284 IF GA=28 THEN 302
286 IF GA>28 THEN C$(1)=" 29":W=W
    +1
288 IF GA>29 THEN C$(2)=" 30":W=W
    +1
290 IF GA>30 THEN C$(3)=" 31":W=W
    +1
292 PRINT"[HOME]":B=5:GOSUB 308:P
    =33
294 FOR V=1 TO W
296 IF FL(V+28)=1 THEN PRINT TAB(
    P)REV$C$(V):PRINT:GOTO 300
298 PRINT TAB(P)C$(V):PRINT
300 NEXTV
```

```
302 GET A$:IF A$="" THEN 302
304 FI$=ME$+AN$
306 GOTO 346
308 B$=LEFT$(GI$,B)
310 PRINTB$:RETURN
312 P=P+4:RETURN
314 CA=CA+1
316 IF CA>6 THEN CA=0
318 IF CA<0 THEN CA=6
320 PRINT:PRINT:PRINT TAB(2);NG$(
    CA);SPC(2)
322 OG$=OG$+STR$(CA):RETURN
324 PRINT:PRINT:PRINT TAB(2)"[GIA
    LLO]GIORNI IN [RVS] REVERSE [
    RVOFF] : APPUNTAMENTO[BIANCO]
    "
326 PRINT:PRINT"[RVS]
                                [R
    VOFF]"
328 FOR S=0 TO 14
330 PRINT"[RVS] [RVOFF]
                                [R
    VS] [RVOFF]"
332 NEXTS
334 PRINT"[RVS]
                           [RVOFF]"
336 PRINT:PRINT TAB(6)"PREMI UN T
    ASTO PER CONTINUARE"
338 PRINT"[HOME]":RETURN
340 REM ************************
342 REM **   MENU PRINCIPALE   **
344 REM ************************
346 POKE 646,1:POKE 53280,6:POKE
    53281,6
348 PRINT"[CLEAR]":PRINT TAB(8)ME
    $(0)
350 FOR J=1 TO 8:PRINT:PRINT TAB(
    9)ME$(J):NEXTJ
352 GET SC$:IF SC$="" THEN 352
354 X=VAL(SC$)
356 IF X<1 OR X>8 THEN 360
358 ON XGOTO 196,386,534,462,484,
    234,370
360 PRINT TAB(11)"[DOWN][GIALLO][
    RVS] NON PREVISTO [RVOFF][BIA
    NCO]"
362 FOR A=0 TO 400:NEXTA:GOTO 346
364 REM ************************
366 REM **   FINE PROGRAMMA    **
368 REM ************************
370 PRINT"[CLEAR]":B=7:GOSUB 308
372 PRINT"HAI MEMORIZZATO I DATI?
     SE SI PREMI 'F'"
374 GET F$:IF F$="" THEN 374
376 IF F$="F" THEN PRINT"[CLEAR]"
    :END
378 GOTO 346
```

```
380 REM ************************
382 REM ** INSERIMENTO APPUNTI *
384 REM ************************
386 PRINT"[CLEAR]":PRINT TAB(8)ME
    $(0)
388 GG$="00":B=4:GOSUB 308
390 PRINT"PER QUALE DATA DESIDERI
     FARE APPUNTI:"
392 PRINT:PRINT TAB(4)GG$;"/";ME$
    ;"/";AN$;
394 INPUT "[10 LEFT]";G$
396 GG$=LEFT$(G$,2)
398 GA=VAL(MID$(XX$,ME*2-1,2)):GG
    =VAL(GG$)
400 IF GG<1 OR GG>GA THEN GG$="":
    GOTO 346
402 FL(GG)=1
404 IF GG=1 OR GG=8 OR GG=15 OR G
    G=22 OR GG=29 THEN D=1
406 IF GG=2 OR GG=9 OR GG=16 OR G
    G=23 OR GG=30 THEN D=2
408 IF GG=3 OR GG=10 OR GG=17 OR
    GG=24 OR GG=31 THEN D=3
410 IF GG=4 OR GG=11 OR GG=18 OR
    GG=25 THEN D=4
412 IF GG=5 OR GG=12 OR GG=19 OR
    GG=26 THEN D=5
414 IF GG=6 OR GG=13 OR GG=20 OR
    GG=27 THEN D=6
416 IF GG=7 OR GG=14 OR GG=21 OR
    GG=28 THEN D=7
418 V1=VAL(MID$(OG$,D*2,1))
420 PRINT"[CLEAR]":PRINT TAB(8)ME
```

```
   $(0)
422 B=1:GOSUB 308
424 PRINT TAB(5)NG$(V1)GG;NM$(ME)
    ;AN
426 GOSUB 514
428 B=6:GOSUB 308
430 PRINT TAB(2)
432 GET P$:IF P$="" THEN 432
434 IF P$=CHR$(136) THEN 452
436 IF P$=CHR$(20) THEN 432
438 IF P$=CHR$(13) THEN 432
440 PRINTP$;
442 AP$=AP$+P$
444 IF LEN(AP$)=35 THEN PRINTCHR$
    (13):PRINT TAB(2)
446 IF LEN(AP$)=70 THEN GOSUB 454
448 IF Q=3 THEN 458
450 GOTO 430
452 AP$(GG,Q)=AP$:Q=0:GOTO 458
454 AP$(GG,Q)=AP$:Q=Q+1:AP$=""
456 PRINTCHR$(13):PRINT TAB(2):RE
    TURN
458 AP$(0,0)="":FOR I=1 TO 31:AP$
    (0,0)=AP$(0,0)+STR$(FL(I)):NE
    XTI
460 AP$="":Q=0:GOTO 346
462 REM ************************
464 REM **    ARCHIVIAZIONE    **
466 REM ************************
468 OPEN 5,8,5,"@0:"+FI$+",S,W"
470 FOR D=0 TO 31
472 FOR T=0 TO 2
474 PRINT#5,AP$(D,T)
476 NEXTT
478 NEXTD
480 FOR E=0 TO 5:PRINT#5,"**":NEX
    TE
482 CLOSE 5:GOTO 346
484 REM ************************
486 REM **    LETTURA DATI     **
488 REM ************************
490 D=0:T=0:H=0
492 OPEN 5,8,5,"0:"+FI$+",S,R"
494 FOR T=0 TO 2
496 INPUT#5,TT$:IF TT$="**" THEN
    T=3:GOTO 500
498 AP$(D,T)=TT$
500 NEXTT:D=D+1
502 IF ST=0 THEN 494
504 CLOSE 5
506 FOR G=1 TO 62 STEP 2
508 H=H+1
510 FL(H)=VAL(MID$(AP$(0,0),G,1))
512 NEXTG:GOTO 346
514 PRINT:PRINT"[RVS]
                                [R
    VOFF]"
516 FOR S=0 TO 14
518 PRINT"[RVS] [RVOFF]
                                [R
    VS] [RVOFF]"
520 NEXTS
522 PRINT"[RVS]
                            [RVOFF]"
524 PRINT:PRINT TAB(6)"PREMI UN T
    ASTO PER CONTINUARE[HOME]"
526 RETURN
528 REM ************************
530 REM **  VISUALIZZAZIONE    **
532 REM ************************
534 PRINT"[CLEAR]":PRINT TAB(8)ME
    $(0)
536 B=3:GOSUB 308
538 PRINT"QUALE GIORNO DESIDERI C
    ONTROLLARE";:INPUT F
540 PRINT"[CLEAR]":PRINT TAB(8)ME
    $(0)
542 B=1:GOSUB 308
544 PRINT TAB(5)NG$(V1)F;NM$(ME);
    AN
546 GOSUB 514
548 B=6:GOSUB 308
550 FOR L=1 TO 68 STEP 34
552 AS$=MID$(AP$(F,Q),L,35)
554 PRINT:PRINT TAB(2)AS$
556 NEXTL
558 Q=Q+1
560 IF Q=3 THEN Q=0:GOTO 564
562 GOTO 550
564 GET A$:IF A$="" THEN 564
566 GOTO 346
```

IL COMMODORE 64 AL MICROSCOPIO

di **Marco De Rosa**

parte 2ª

In questa puntata troverete la descrizione delle locazioni che vanno dalla 139 alla 197. Come di consueto le righe in neretto nella mappa completa sono descritte in dettaglio nel testo.

(144): Byte di status

Questa locazione contiene la variabile di sistema STATUS. In BASIC ST è una parola riservata e non può essere usata come nome di variabile. Il suo valore cambia se ci sono dei problemi nel caricamento da disco o da nastro, e può quindi essere utilizzata come flag.
Supponiamo per esempio di dover leggere un file sequenziale da disco, senza sapere a priori la sua lunghezza. Una tipica routine di lettura potrebbe essere la seguente:

```
100 OPEN 2, 8, 2, "0: NOME, S,
R"
110 INPUT #2, P (I)
120 I=I+1
130 IF ST=64 THEN 200
140 GOTO 110
200 CLOSE 2
210 STOP
```

L'istruzione della **riga 130** si accorge se il file è finito e interrompe il ciclo di lettura. Possiamo quindi dire che l'uso di ST può sostituire l'istruzione di
ON ERROR GOTO assente nel BASIC del CBM 64, nel caso di istruzioni di LOAD.
In figura 1 potete vedere i valori assunti da ST al verificarsi delle configurazioni di errore.

(152): Numero di file aperti

Questa locazione contiene il numero dei file contemporaneamente aperti. Il numero massimo è dieci, dopodiché il BASIC bloccherà il programma comunicandovi un "TOO MANY FILES OPEN ERROR". Per ovviare a questo inconveniente nel caso vi troviate con un numero di files aperti vicino a dieci potete aggiungere le seguenti istruzioni prima di aprirne un nuovo:

```
100 IF PEEK (152)=10 THEN
1000
110 OPEN 2, 8, 2, "0: NOME, S,R"
.
.
.
1000 REM TROPPI FILES APERTI
1010? "HAI GIA' 10 FILES A-
PERTI"
.
.
```

(153-154): Dispositivo di I/O

Queste locazioni contengono il numero dei due dispositivi di Input/Output operanti attualmente. I valori di default sono 0 e 3 che corrispondono rispettivamente alla tastiera e allo schermo. Il valore della seconda locazione può essere cambiato da BASIC usando l'istruzione CMD dopo un OPEN. La sintassi è la seguente:

OPEN i fn, dn, [sa]
CMD ifn
Istruzione BASIC

dove ifn=numero di file logico, dn=numero di device (dispositivo), sa=indirizzo secondario. L'ifn può essere un numero da 1 a 255 con alcune limitazioni che dipendono dalla periferica usata (consultate la Programmer's Reference Guide). L'sa è opzionale (vedi **locazione 185** in questo articolo). Il dn dipende dalla periferica usata. In figura 2 potete trovare la lista dei numeri di device. Se per esempio volete mandare un listato sulla stampante (il cui numero è normalmente 4) potete scrivere:

OPEN 3,4: CMD3: LIST

Ricordatevi di chiudere il file con il seguente comando:

PRINT# 3: CLOSE 3

(da 160 a 162): Jiffy clock

Queste locazioni contengono il cosidetto "Jiffy" clock, cioè l'orologio che controlla le operazioni eseguite dal computer. Il più piccolo intervallo di tempo misurabile (Jiffy) è 1/60 di secondo.
Il funzionamento è il seguente.
- Ogni sessantesimo di secondo viene aumentato di 1 il valore della **locazione 162.**
- Quando questo giunge a 255, viene riportato a zero e viene aumentato di 1 il valore della **161.**
- Quando anche questa giunge a 255, viene riportata a zero e viene aggiunto 1 alla locazione **160.**
- Quando anche questa raggiunge 255 le tre locazioni vengono riportate a zero.

Il periodo totale dell'orologio è quindi di 16.581.375 sessantesimi di secondo, cioè 77 ore.
Il BASIC gestisce due variabili di sistema, TI e TI$, controllate dal jiffy clock:
- TI parte dal valore "000000" quando accendete il computer e viene aggiornata ogni sessantesimo di secondo fino a raggiungere il valore "999999", dopodiché viene riportata a zero.
- TI$ parte anch'essa da "000000" ma viene aggiornata ogni secondo. Essa contiene l'ora nel formato "OOMMSS", dove OO rappresenta le ore, MM i minuti e SS i secondi. Se per esempio volete mettere l'ora alle 16 e 35 del pomeriggio dovrete scrivere:

TI$ = "163500"

In ogni momento potete accedere all'orologio con una routine del tipo

100 PRINT "[CLR]"

```
110 PRINT "[HOME]"; TI$
120 GOTO 110
```

L'orologio è molto preciso fino a che non eseguite delle operazioni da disco o da cassetta.

(167-168-169-170-171-180 -181 -182-189): Area RS-232

Il Commodore 64 possiede delle routine di sistema che permettono di controllare da BASIC tutti i dispositivi RS-232. Queste locazioni sono usate direttamente dai dispositivi attraverso le routines suddette. Esse contengono anche dati relativi allo svolgimento delle operazioni di cassetta e del bus seriale. Durante queste operazioni l'interfaccia RS-232 non è quindi utilizzabile.
Per informazioni dettagliate consultate la Programmer's Reference Guide.

(174-175): Fine buffer in SAVE e LOAD

Queste locazioni contengono l'indirizzo da cui le routines di LOAD e SAVE cominciano a caricare o salvare i programmi BASIC. Se voi cambiate le locazioni **45** e **46** (inizio del testo BASIC) dovete assolutamente aggiornare anche queste con gli stessi valori.

(185): Indirizzo secondario corrente

Questa locazione contiene il numero dell'indirizzo secondario del file corrente. Questo numero viene stabilito nell'operazione di OPEN relativa a quel file. In figura 3 potete vedere gli indirizzi secondari utilizzabili per ogni device.
Se per esempio volete stampare un testo in carattere Maiuscolo/Minuscolo invece di default Maiuscolo/Grafica dovrete comunicarlo alla stampante con l'istruzione:

```
OPEN 3, 4, 7
```

seguita dalle istruzioni di PRINT#3 necessarie.

Posizione Bit Status	Valore Numerico	Lettura Cassette	Seriale/RW	VerifY o Load Registratore
Scrittura 0	1		Time Out	
Lettura 1	2		Time Out	
2	4	Blocco corto		Blocco corto
3	8	Blocco Lungo		Blocco Lungo
4	16	Err. di Lettura		Qualsiasi Incongruenza
5	32	Err. Cecksum		Errore di Checksum
6	64	Fine File	*EOI line	
7	128	Fine Nastro	Device Non Presente	Fine Nastro

Figura 1: i valori assunti da ST al verificarsi di configurazioni di errore.

Device Number	Periferica
0	Tastiera
1	Cassetta
2	Modem
3	Schermo
4 o 5	Stampante
da 8 a 11	Disk Drivers

Figura 2: i numeri di device.

Device	Numero Device	Indirizzo Secondario	Descrizione
Cassette	1	0=Input 1=Output 2=Output	Nome del file Fine del nastro
Modem	2	0	Registro di controllo
Stampante	4 o 5	0=maiuscole/grafica 7=maiuscole/minuscole	Testo in stampa
Disco	da 8 a 11	2-14 canale dati 15= canale comandi	Drive#, tipo file, comando di lettura o scrittura

Figura 3: indirizzi secondari utilizzabili per ogni device.

HEX	DEC	DESCRIZIONE
008B-008F	139-143	Valore del "seme" della funzione RND.
0900	144	**Variabile di sistema STATUS. Il suo valore di default é zero. In BASIC questo nome é riservato al sistema. Il valore di ST cambia se ci sono dei problemi nel caricamento di dati da disco o dal nastro.**
0091	145	Flag: Tasto di STOP e di RVS. Il suo valore di default è 255. L'istruzione POKE 145,127 simula l'azione del tasto RUN/STOP.
0092	146	Costante di tempo per il nastro.
0093	147	Flag: 0=LOAD, 1=VERIFY. Memorizza l'ultima operazione di LOAD o VERIFY del disco o del nastro.
0094	148	Flag: buffer dei caratteri in uscita dal bus seriale.
0095	149	Carattere bufferizzato per il bus seriale.
0096	150	Numero di sincronizzazione per cassetta.
0097	151	Area per dati temporanei.
0098	152	**Numero di files aperti e indici alla tavola dei files. Potete tenere aperti fino a dieci files contemporaneamente. Se cercate di superare questo limite il BASIC risponderà con il messaggio "TOO MANY FILES OPEN ERROR".**
0099	153	**Dispositivo di input di default. Normalmente è zero (Tastiera).**
009A	154	**Dispositivo di output (CMD) di default. Normalmente è tre (Schermo).**
009B	155	Carattere di parità del nastro.
009C	156	Flag: Byte ricevuto dal nastro.
009D	157	Flag: 128= Modo diretto, 0= modo programma.
009E	158	Errore logico da nastro. Passo 1.
009F	159	Errore logico da nastro. Passo 2.
00A0-00A2	160-162	**JIFFY Clock.**
00A3-00A4	163-164	Area dati temporanei.
00A5	165	Conto alla rovescia per il sincronismo di cassetta.
00A6	166	Pointer: buffer di I/O da nastro.
00A7	167	**Bits di input per RS-232/Area temporanea per cassetta.**
00A8	168	**Contatore dei bits di input per RS-232/Area temporanea per cassetta.**
00A9	169	**Flag: controlla il bit di start per RS-232.**
00AA	170	**Buffer di input per byte in RS-232/Area temporanea per cassetta.**
00AB	171	**Parità di input per RS-232/Contatore per cassetta.**
00AC-00AD	172-173	Pointer: buffer per cassetta/Scroll delo schermo.

HEX	DEC	DESCRIZIONE
00AE-00AF	174-175	**Indirizzo fine programma per operazioni di SAVE e LOAD.**
00B0-00B1	176-177	Costanti di tempo per il nastro.
00B2-00B3	178-179	Pointer: inizio del buffer di cassetta.
00B4	180	**Contatore bit di uscita per RS-232/Area temporanea per cassetta.**
00B5	181	**Prossimo bit da inviare per RS-232/ Flag per End Of Tape**
00B6	182	**Buffer per byte in uscita per RS-232.**
00B7	183	Lunghezza del nome del file corrente, cioè il numero di caratteri nel nome del file stesso. E' la stessa cosa che scrivere PRINT(LEN(NOME$)), dove NOME$ é il nome del file.
00B8	184	Numero del file logico corrente.
00B9	185	**Indirizzo secondario corrente.**
00BA	186	Numero del dispositivo (device) corrente.
00BB-00BC	187-188	Pointer: nome del file corrente.
00BD	189	**Parità di uscita per RS-232/ Zona temporanea per cassetta.**
00BE	190	Contatore di blocchi del nastro in lettura o scrittura.
00BF	191	Buffer per porta seriale.
00C0	192	Interruttore per il motore della cassetta.
00C1-00C2	193-194	Indirizzo di partenza per I/O.
00C3-00C4	195-196	Usati per l'immagazzinamento temporaneo di informazioni durante il caricamento da cassetta.

PROGRAMMAZIONE STRUTTURATA

di Mariangela Guardione

terza parte

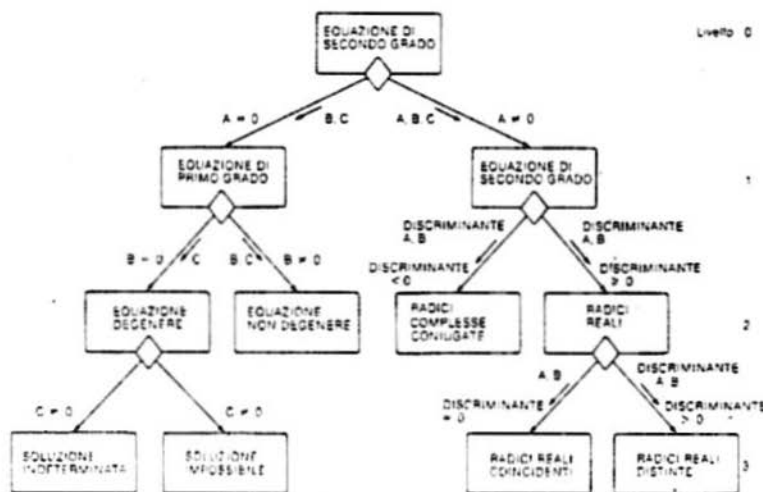



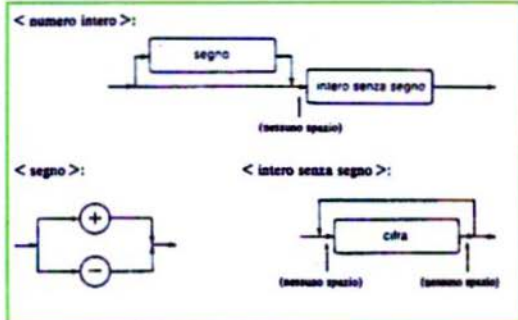
Figura 1: rappresentazione grafica del programma

In questa puntata continueremo il nostro viaggio attraverso la conoscenza delle istruzioni BASIC, da un punto di vista strutturato. Prima però d'introdurre nuove istruzioni esaminimamo i tre tipi di dati elementari, cioè quelli che possono essere elaborati e/o trasformati per mezzo di operatori o funzioni. Sono:

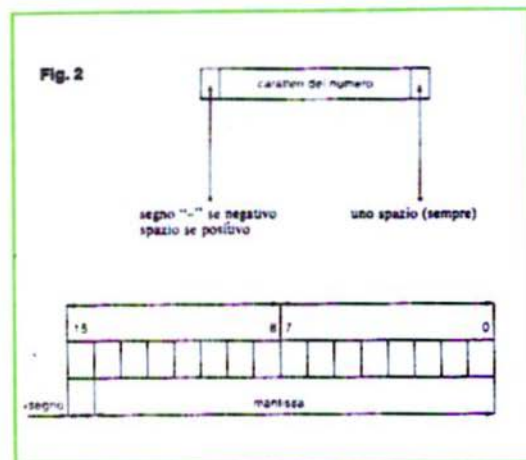
- NUMERI INTERI
- NUMERI REALI
- STRINGHE DI CARATTERI-

(che saranno oggetto della prossima puntata).

Numeri interi

Un numero intero è una grandezza numerica che viene rappresentata senza virgola, e il cui diagramma sintattico è dato dallo schema che segue.

I numeri interi in memoria centrale vengono rappresentati con 16 bit (cioè 2 bytes), il più significativo dei quali è quello del segno. I numeri negativi

sono rappresentati in complemento a due (questo è un artificio che permette al calcolatore di effettuare l'operazione di sottrazione che altrimenti non sarebbe in grado di eseguire).

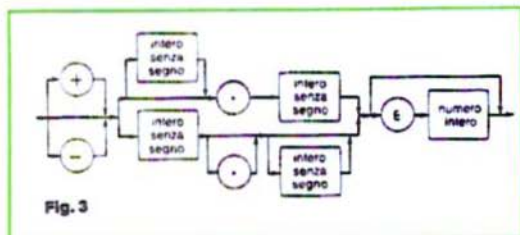
Fig. 2

I massimo intervallo possibile per gli interi è quindi:

min. int. = -32768
max. int. = +32767.

Lo zero viene considerato positivo ed è quindi per questo che i numeri negativi posseggono una unità in più in valore assoluto.

Numeri reali

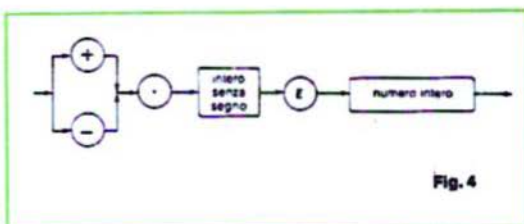
Il numero reale viene rappresentato tramite il diagramma sintattico nel modo illustrato nello schema seguente:

Fig. 3

dove "E" sta per moltiplicato 10 elevato a potenza n, mentre il punto rappresenta, nella notazione anglosassone, la virgola decimale.

Il "numero intero" in genere non supera le due cifre. Esempio:

24.12.10 (-15)=24.12.E-15; 1.5; 12

Un numero reale può essere rappresentato in una forma normalizzata secondo lo schema che segue.

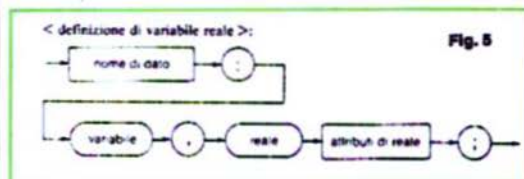
Fig. 4

Un numero reale che non si presenti in forma normalizzata, può essere trasformato in quest'ultima nel modo seguente: si sposta il punto decimale verso sinistra di tante cifre quante dovrà essere l'incremento di unità sul valore dell'esponente. Esempio:

50.12E-17

portato in forma normalizzata diventa

0.5012E-15.

A questo punto passiamo alla definizione di "variabile reale" che in pseudo-codifica ha la rappresentanza riportata nello-schema seguente.

Fig. 5

Espressione reale 2 e espressione reale 1 rappresentano gli estremi dell'intervallo min. reale e max. reale. Se essi vengono specificati rappresentano, a livello di codifica, i limiti che sono imposti dal linguaggio, vengono anche usati nella fase iniziale di analisi del progetto quando non si ha ancora idea del limite inferiore o superiore della variabile considerata, o ancora quando questi due limiti possono delimitare un'intervallo il più ampio possibile.
Precisione indica il numero di cifre significative che devono essere prese in considerazione.
Espressione reale e espressione intera sono espressioni algebriche il cui risultato è rispettivamente un numero reale o intero.

Rappresentazione dei numeri reali in BASIC

Nel linguaggio di programmazione che stiamo esaminando, i numeri reali sono compresi nell'intervallo:

NUMERI NEGATIVI: da -10 +38 a -10 -38
NUMERI POSITIVI: da 10 -38 a 10 +38

Per quanto riguarda la precisione (e qui si intende il BASIC standard e non quello specifico del Commodore), vi sono due possibilità:

- singola precisione
- doppia precisione

Singola precisione

In questa rappresentazione vengono memorizzate non più di 7 cifre e stampate non più di 6. Come avviene per i numeri interi, in fase di stampa si ha sempre uno spazio bianco dopo la cifra più a destra e uno spazio bianco a sinistra del numero se positivo.
Sul Commodore si ha solamente la singola precisione con 9 cifre.

Doppia decisione

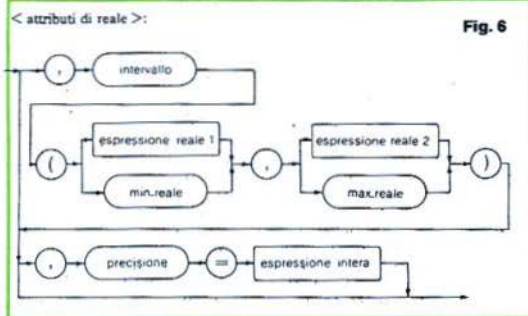
In questa maniera vengono memorizzate e stampate non più di 16 cifre significative. Per quanto riguarda la stampa si hanno le stesse modalità viste per la singola precisione.
Nello "schema 6" viene riportata la rappresentazione dei numeri reali in memoria centrale:

A proposito della rappresentazione in BASIC dei numeri, bisogna precisare che in matematica i "numeri interi" sono un sottoinsieme (cioè una parte) dei "numeri reali".
Nei linguaggi di programmazione, invece, succede che questi due insiemi di numeri vengono trattati come classi distinte a causa della loro diversa rappresentazione e per il modo diverso con cui le istruzioni macchina generate dal linguaggio di codifica trattano gli elementi di queste due classi in fase di utilizzo degli stessi operatori aritmetici.
In matematica l'insieme dei numeri reali presenta uno spettro continuo di infiniti valori, mentre nella memoria di un calcolatore questi vengono rappresentati-con numero finito di cifre. Tutto questo porta ad avere un numero finito di valori che possono essere rappresentati con uno spettro discreto di valori che porta, una volta scelta la singola oppure la doppia precisione, ad essere implicitamente determinato il minimo errore relativo che può essere compiuto.

Fig. 7

In applicazioni nel campo del calcolo numerico condotte con metodi interattivi, il numero finito di cifre genera errori di troncamento che si propagano verso le cifre più significative a cui è sottoposto uno stesso numero.
Quindi ove sia possibile, è sempre opportuno confrontare una differenza (o un suo valore assoluto) con un numero molto piccolo; il termine "piccolo" va scelto in maniera opportuna (differente cioè caso per caso), in quanto esso esprime da quale ordine di grandezza in poi non abbia più senso, in situazioni particolari, tentare di apprezzare la differenza tra due valori. Passiamo ora alla descrizione degli operatori e delle funzioni in BASIC:

Operatori

In BASIC e negli altri linguaggi di programmazione le priorità degli operatori aritmetici nelle espressioni sono le medesime rispetto a quelle delle espressioni algebriche.

Operatore	Commento	Esempio
–	negazione	– X
^	elevamento a potenza	X ^ Y
•,/	moltiplicazione, divisione	X • Y, X/Y
\	divisione intera	X \ Y
MOD	modulo	X MOD Y
+, –	addizione, sottrazione	X + Y, X – Y

Fig. 8.1

Per variare l'ordine di esecuzione delle operazioni devono essere utilizzate le parentesi tonde come avviene in algebra.

Funzioni

Le funzioni sono quegli operatori che, applicati all'argomento racchiuso fra parentesi, restituiscono un valore che, se non viene specificatamente precisato, è reale e in singola precisione.
Qui di seguito vengono riportati i diagrammi sintattici delle funzioni più importanti nel BASIC, in cui per brevità di scrittura si indicherà con X, Y: (espressioni reali) e I, J: (espressioni intere)

Riprendiamo ora ad esaminare le altre istruzioni BASIC.

Fig. 8.3

Fig. 8.4

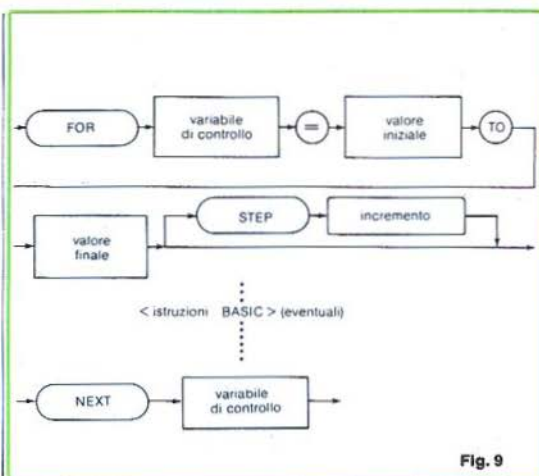
Fig. 8.5

Istruzione di costruzione di LOOP: FOR... NEXT

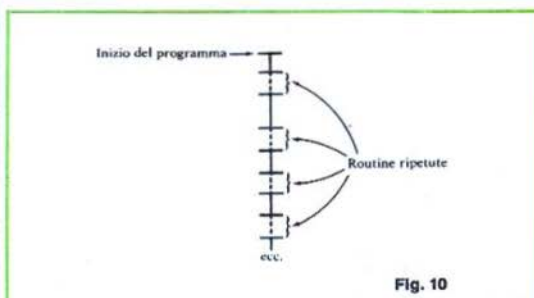
Abbiamo fin qui potuto ammirare la versatilità di un calcolatore, rappresentata non solo dalla capacità di eseguire in breve tempo una particolare sequenza di calcoli, quanto nella possibilità di prendere delle decisioni logiche.
Un'altra operazione che è spesso richesta nei programmi è la ripetizione (looping). Si tratta cioè di ripetere una parte di istruzioni un certo numero di volte, oppure finché non sia stata soddisfatta una certa condizione. Il "loop" è quella parte di programma che viene ripetuta in modo da evitare che le istruzioni comprese nella porzione di programma che viene ripetuto siano scritte più di una volta. Tutto questo in BASIC si ottiene utilizzando il FOR... NEXT. La sintassi corretta di questa istruzione può essere:

FOR a=n1 TO n2 STEP n3

In questa prima parte si specifica quante volte il loop sarà eseguito. In questa istruzione è compresa una variabile ordinaria (a) alla quale una volta assegnato un valore iniziale (n1) e uno finale (n2), serve a determinare il numro di esecuzioni desiderate in funzione del valore di incremento (n3).
Per determinare un "loop" viene utilizzata l'istruzione NEXT seguita dalla variabile che appare nel corrispondente enunciato FOR... TO.

Fig. 9

Procedure (subroutines)

Quando si scrivono dei pro-grammi strutturati con un certo numero di righe, si possono avere delle sequenze d'istruzioni (det-te procedure o subroutines) che vengono usate ripetutamente. Come esempio si potrebbe pensare ad un programma in cui si devono ripetere diverse volte le tre istruzioni che costituiscono il loop FOR... NEXT, descritto in precedenza. Tutto questo porterebbe ad un inutile spreco di tempo, e ancora più grave sarebbe lo spreco di spazio di memoria. Ciò può essere illustrato nello schema che segue.

Fig. 10

Per risolvere questi problemi si potrebbe scrivere separatamente queste righe di programma e richiamarle ogni volta che necessita; questo gruppo di istruzioni si chiama "subroutine".
Tutto questo però porta ad un inconveniente: saltare dal programma alla subroutine è semplice dato che essa comincia ad un numero di riga determinato, mentre il punto di ritorno al programma principale varia di volta in volta.

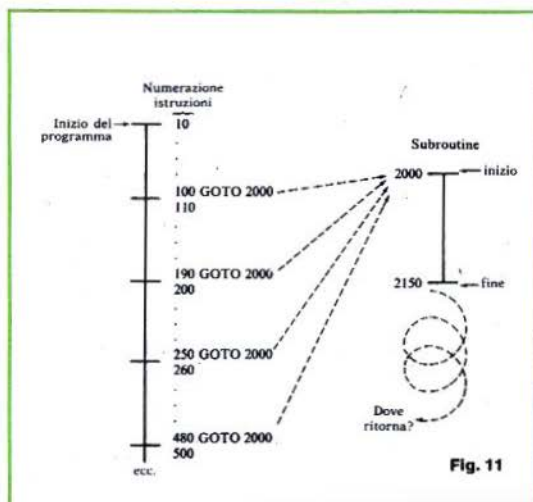



Fig. 11

Per risolvere tutto questo occorre utilizzare una nuova istruzione.

Istruzione GOSUB

Questa istruzione presenta un comportamento analogo al GOTO, con l'aggiunta però del numero di riga alla quale il calcolatore deve ritornare. Il GOSUB, cioè, fa in modo che l'elaboratore "ricordi", in qualche modo, l'indirizzo dell'istruzione successiva alla medesima GOSUB.

A questa istruzione è abbinata un'altra, il RETURN, la quale induce a ritornare esattamente all'istruzione seguente a quella che ha generato l'esecuzione della subroutine; quel punto che GOSUB aveva memorizzato in partenza.
Bisogna però anche dire che le subroutine possono presentarsi annidate: possono cioè ciascuna chiamarsi a vicenda. Non è necessario per fare questo utilizzare altre istruzioni, basta semplicemente fare un salto ad un'altra subroutine utilizzando l'istruzione GOSUB.
Bisogna però ricordare che ogni subroutine deve terminare con l'istruzione RETURN, che segnala appunto al calcolatore la sua fine.
Lo schema riporta la rappresentazione grafica delle GOSUB.

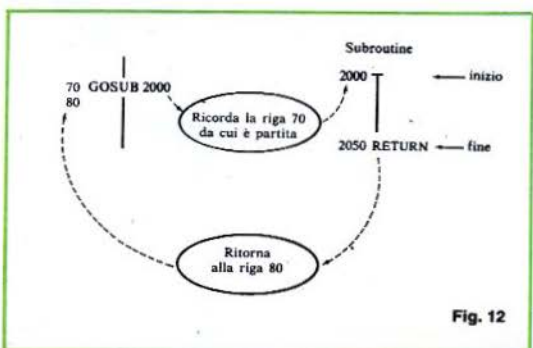



Fig. 12

Istruzione ON... GOSUB e-ON... GOTO

Queste istruzioni permettono delle scelte multiple; in base al valore della variabile scritta dopo ON, il calcolatore sceglie a quale linea andare (la definizione di queste linee saranno poste proprio dopo l'istruzione GOTO e GOSUB).
Le istruzioni hanno la seguente struttura:

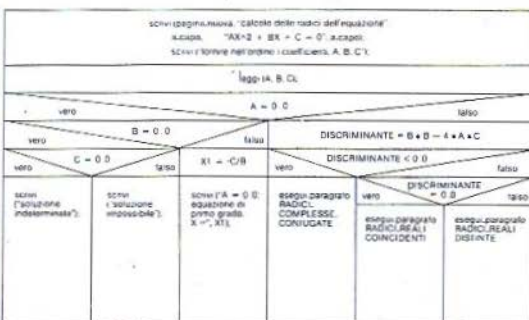
ON X GOTO L1, L2, L3, L4,...
ON X GOSUB L1, L2, L3, L4,...

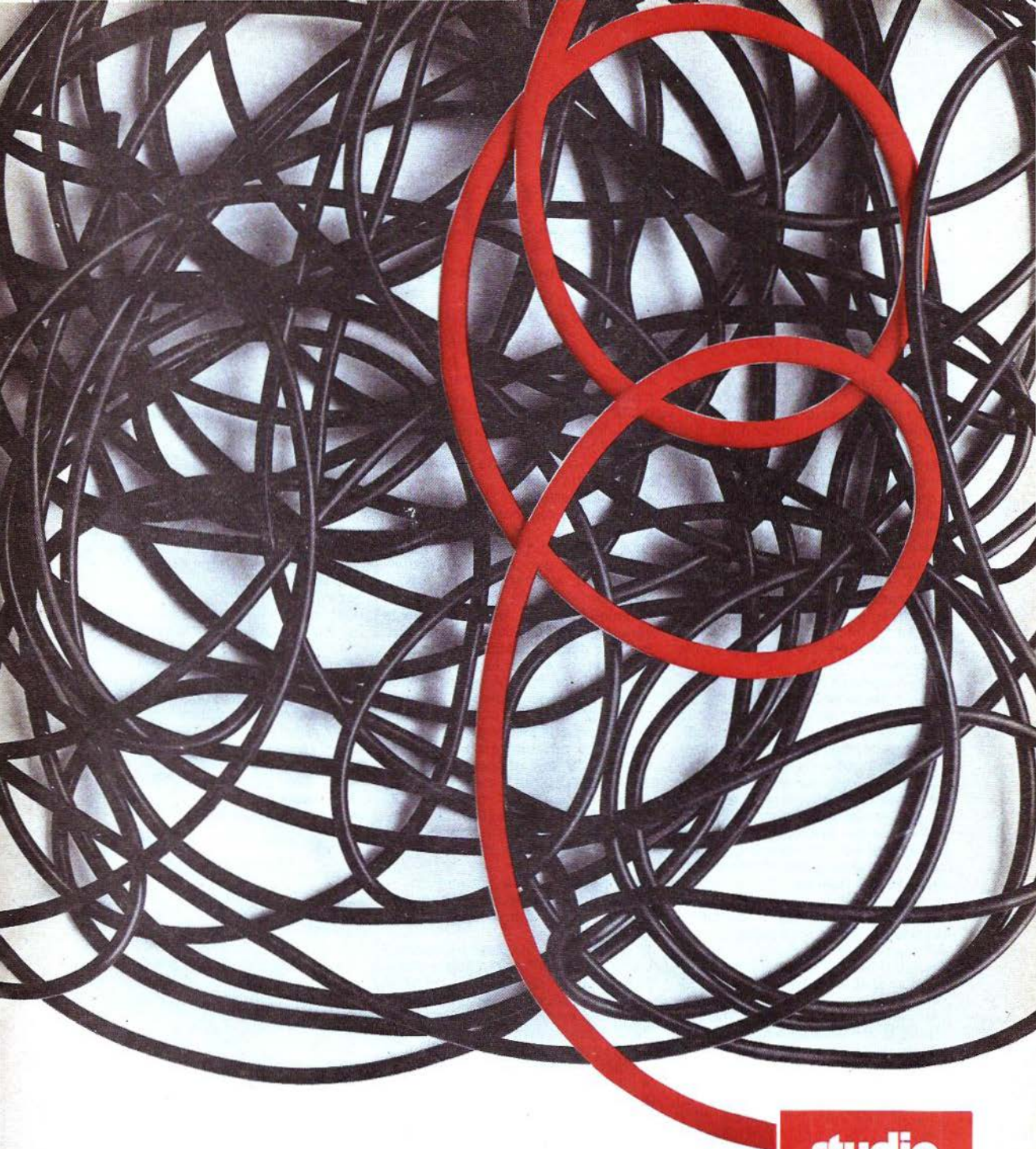
Quando il calcolatore incontra tale istruzione, verifica il valore di X e decide dove andare:

- se X=1 allora salta a L1, intendendo con L1 il primo numero di linea posto dopo il GOTO (oppure dopo il GOSUB)
- se X=3 allora salta L3, terzo numero di linea
- prosegue così per tutti i numeri di linea posti dopo il GOTO oppure dopo il GOSUB.

Nel BASIC standard, se il valore di X è minore o uguale a zero (oppure maggiore del numero di linee presenti) il calcolatore ignora l'istruzione e prosegue dalla linea seguente. Nel BASIC del Commodore invece se il valore di X è minore a zero si comporta esattamente come per il valore uno.
Lo schema riporta la loro rappresentazione grafica.

Fig. 14

Per concludere questa puntata riportiamo come esercizio per l'applicazione delle istruzioni fin qui esaminate, il programma che calcola le "radici di un'equazione di secondo grado"; come per gli altri programmi è stato riportato non solo il listato, ma anche la sua rappresentazione grafica.

```
PROGRAMMA : EQUAZ PRSTR 3 PR

------*---------------- - - -
VAR.  ! LINEA DEL PROGRAMMA
------*---------------- - - -
      !
A     !    29    30    41    51    52    61
      !    69    70
B     !    29    32    37    41    51    61
      !    69    70
C     !    29    34    37    41
D     !    41    42    43    50    52    69
      !    70
X1    !    37    38    51    54    55    61
      !    63    69    72
X2    !    52    54    55    70    72
```

```
10 REM ***********************
11 REM *  CALCOLO EQUAZIONE   *
12 REM *   DI SECONDO GRADO   *
13 REM *         DI           *
14 REM * GUARDIONE MARIANGELA *
15 REM ***********************
16 REM *DATI IN INGRESSO:A,B,C*
17 REM *DATI IN USCITA        *
18 REM *X1=UNICA RADICE REALE *
19 REM *COINCIDENTE           *
20 REM *X1,X2=COPPIA DI RADICI*
21 REM *REALI DISTINTE        *
22 REM *X1=PARTE REALE,X2=PARTE
23 REM *IMMAGINARIA NEL CASO  *
24 REM *DI RADICI COMPLESSE   *
25 REM *INIZIO ELABORAZIONE   *
26 REM ***********************
27 PRINT"[CLEAR][2 DOWN]RISOLUZI
   ONE DELL'EQUAZIONE"
28 PRINT"DI TIPO AX↑2+BX+C=0[2 D
   OWN]"
29 INPUT "FORNIRE I COEFFICIENTI
    A,B,C";A,B,C
30 IF A=0 THEN 32
31 GOTO 41
32 IF B=0 THEN 34
33 GOTO 37
34 IF C=0 THEN PRINT"[DOWN]SOLUZ
   IONE INDETERMINATA":GOTO 36
35 PRINT"[DOWN]SOLUZIONE IMPOSSI
   BILE"
36 PRINT:GOTO 39
37 X1=-C/B
38 PRINT"[DOWN]EQUAZIONE DI PRIM
   O GRADO; RADICI COINCIDENTI X
   =";X1
39 GOTO 45
40 PRINT"[DOWN]EQUAZIONE DI SECO
   NDO GRADO"
41 D=B↑2-4*A*C
42 IF D<0 THEN GOSUB 50:GOTO 45
43 IF D=0 THEN GOSUB 61:GOTO 45
44 GOSUB 69:GOTO 45
45 END
46 REM *********************
47 REM * SUBROUTINE RADICI *
48 REM * COMPLESSE         *
49 REM *********************
50 D=-D
51 X1=-B/(2*A)
52 X2=SQR(D)/(2*A)
53 PRINT"[DOWN]RADICI COMPLESSE
   CONIUGATE"
54 PRINTX1;"+I";X2
55 PRINTX1;"-I";X2;"[DOWN]"
56 RETURN
57 REM *********************
58 REM * SUBROUTINE RADICI *
59 REM * REALI COINCIDENTI *
60 REM *********************
61 X1=-B/(2*A)
62 PRINT"[DOWN]RADICI REALI COIN
   CIDENTI"
63 PRINT X1;" ; ";X1;"[DOWN]"
64 RETURN
65 REM *********************
66 REM * SUBROUTINE RADICI *
67 REM * REALI DISTINTE    *
68 REM *********************
69 X1=-B/(2*A)+SQR(D)/(2*A)
70 X2=-B/(2*A)-SQR(D)/(2*A)
71 PRINT"[DOWN]RADICI REALI DIST
   INTE"
72 PRINTX1;" ; "X2;"[DOWN]"
73 RETURN
```

LA STATISTICA

di **Mariangela Guardione**

prima parte

Con questa serie di articoli, che inizia su questo numero della nostra rivista, si vuole esplorare una delle discipline oggi più importanti, ma poco conosciutà: la Statistica. Questa parola molto spesso suscita sensazioni di diffidenza perché associata a qualcosa di complicato e molto difficile, riservata cioè a pochi esperti del campo. In realtà la statistica rappresenta un sistema, cioè un insieme di strumenti, che serve a misurare la realtà. Ad esempio: quando vogliamo misurare la distanza fra la Terra e le stelle, utilizziamo come unità di misura "l'anno luce"; se invece vogliamo misurare un pezzo di stoffa utilizziamo il metro.
Se si devono però misurare le merci che un paese ha esportato o importato, ecco che serviranno delle unità di misura diverse da quelle che utilizziamo normalmente più complesse, meno note e meno facili da applicare: le "misure statistiche". In generale, ogni fenomeno consiste in un insieme di cause e di effetti che nella scienza deterministica è rappresentato dall'evolversi di grandezze che lo descrivono mediante un modello che rappresenta un insieme di variabili adatte a rappresentare le cause (input) e gli effetti (output) e le relazioni che le legano fra loro.
Ad esempio per studiare il moto dei pianeti si utilizza il modello di Keplero, quindi si schematizza il fenomeno fisico attraverso una rappresentazione visiva per poterlo meglio studiare. In natura però non esistono sistemi per i quali sia possibile determinare in maniera univoca gli effetti tramite le relazioni che li legano alle variabili d'ingresso sia perché i sistemi reali sono complessi ed il descriverli da un punto di vista deterministico richiederebbe l'introduzione di un numero grandissimo di variabili e di relazioni.
Per questo motivo si ricorre a modelli semplificati, costituiti da un numero minore di variabili e di relazioni accontentandosi del fatto che tali modelli approssimino l'andamento reale più che riprodurlo esattamente. Quindi il compito della statistica è quello di incasellare la realtà in schemi ben definiti, che però non alterino al sua veridicità.
Abbiamo parlato di "variabili": che cosa sono e che cosa rappresentano?
Una variabile, ad esempio la X, è un simbolo che può assumere qualunque valore all'interno di un predeterminato insieme di valori detto "dominio" della variabile. Se questa variabile può assumere un solo valore viene detta "costante". Una variabile che, in termini teorici, può assumere un qualunque valore fra due valori dati, viene chiamata "variabile continua", in caso contrario viene detta "variabile discreta".
I dati che possono essere descritti tramite variabili discrete vengono detti "dati discreti", mentre gli altri sono "dati continui". Il numero dei bambini di 2000 famiglie è un esempio di dati discreti, mentre l'altezza di 50 studenti è un esempio di dati continui.
Da tutto ciò si può ricavare che le misurazioni danno origine a dati continui, mentre le enumerazioni a quelli discreti.
In statistica la relazione fra variabili viene rappresentata mediante grafici (detti anche "diagrammi"). Vi sono vari tipi di rappresentazioni del genere, che vedremo in seguito in dettaglio e che sono:

- il diagramma a rettangoli
- i diagrammi circolari
- gli istogrammi
- le regressioni

A questo punto incominciamo ad esaminare i più importanti concetti largamente utilizzati in statistica.

Concetto di media

Una media, in genere, viene calcolata perchè permette di riassumere in un'unica cifra un insieme, spesso numeroso, di dati (come per esempio il salario medio di impiegati in un industria). Questa grandezza inoltre permette i confronti fra le misure di due eventi diversi (per esempio il consumo medio di carne e quello di pesce in Italia), o dello stesso fenomeno in tempi diversi (ad esempio la crescita media in altezza degli italiani prima e dopo la seconda guerra mondiale), o in luoghi diversi (come il diverso reddito medio degli italiani e degli americani).
Data quindi una distribuzione di frequenza, cioè un ordinamento tabulare dei dati secondo le classi, i valori medi si dicono anche misure di posizione poiché essi individuano sulla scala delle ascisse, elementi importanti per la stessa distribuzione. I valori medi più usati in statistica sono i seguenti.

- la media aritmetica
- la media geometrica
- la media armonica
- la media quadratica

Media aritmetica

La media aritmetica di un insieme di N numeri X1, X2, X3, ...XN, indicata col simbolo $\bar{X}$, è data da:
$\bar{X}$= (X1+X2+ ...+XN) /N
Se ad esempio vogliamo ottenere la media aritmetica dei cinque numeri 8, 2, 7, 6, 13, dovremmo calcolare.
$\bar{X}$= (8+2+7+6+13+) /5=36/5=7.2
cioè:
$\bar{X}$=7.2
Vi è anche la "media aritmetica ponderata"; questa si ha quando ad N numeri X1, X2, ...XN vengono associati dei pesi w1, w2 ..., wN cioè sono le frequenze con cui si verificano i vari numeri. Essa è data dalla formula:
$\bar{X}$= (w1X1+w2X2+ .....+wNXN) / (w1+w2+ ...wN)
Se ad un esame orale si attribuisce un peso tre volte superiore a quello

attribuito per lo scritto e uno studente ottiene come voto per l'orale 85 mentre per gli scritti 70 e 90 la media dei voti è:
X=((1)*70+(1)*90+(3)*85) / (1+1+3)=83

Media geometrica

La media geometrica di un insieme di N numeri X1, X2, ..., XN è la radice N-esima del prodotto dei numeri:
G= (X1*X2* .....*N) ↑ (1/N)
La media geometrica dei quattro numeri 2, 3, 7, 9 è:
G= (2*3*7*9) ↑ (1/4)=4.409

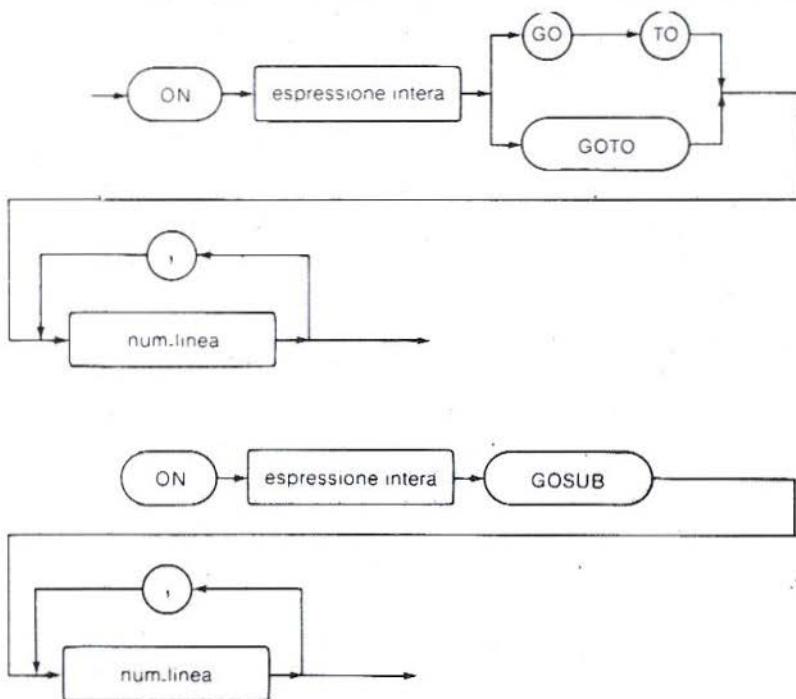
Media armonica

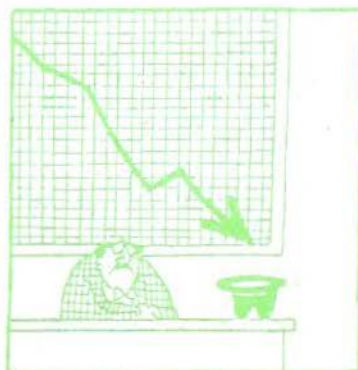
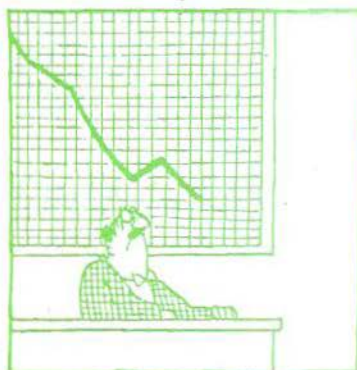
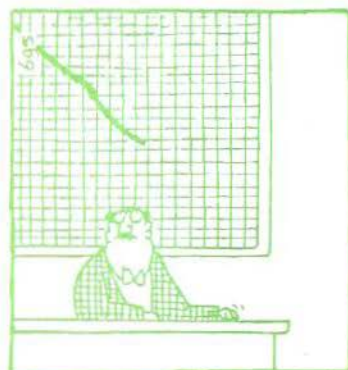
La media armonica di N numeri X1, X2, ...XN è il reciproco della media aritmetica dei reciproci dei numeri stessi cioè:
H=/ (1/N* (1/X1+1/X2+ ...+1/XN))
La media armonica dei tre numeri 4, 8, 12 e
H= 1/ (1/3* (1/4+1/8+1/12))=72/11
Esiste una relazione fra le medie fin qui viste: la media geometrica di un insieme di numeri positivi X1, X2, ..., XN è minore o uguale alla loro media aritmetica, ma maggiore o uguale alla loro media armonica. Questa relazione è rappresentata simbolicamente da:
H<=G<=X
Questi segni di uguaglianza valgono naturalmente se tutti gli N numeri utilizzati per calcolare le medie sono identici.

Media quadratica

La media quadratica di un insieme di numeri X1, X2, ..., XN è definita nel seguente modo:
(X↑2) ↑ (.5)=((X1↑2+X2↑2+ ...+XN↑2) /N)↑ (.5)
La media quadratica dei numeri 1, 3, 5, 7 è:
(X↑2) ↑ (.5) = ((1↑2+3↑2+5↑2+7↑2) /4) ↑ (.5) =4.58
L'ultima grandezza statistica che esaminiamo in questa prima parte prende il nome di "momento": esso serve a mettere in evidenza alcune importanti caratteristiche di un insieme di dati.
Se X1, X2, ..., XN sono gli N valori assunti dalla variabile X, la quantità X↑r= (X1↑r+X2↑R+...+XN↑r) /N
Viene definita: "momento di ordine r". Il momento di primo ordine, r=1, è la media aritmetica X.
Se ad esempio vogliamo trovare i

```
C R O S S   R E F E R E N C E

PROGRAMMA   MEDIA PRG

---------*----------------------------
VAR.     ! LINEA DEL PROGRAMMA
---------*----------------------------
         !
A        !   38    79    80    81    82    83
AP       !   47    84
AS       !   48    85
FC       !   26    29    63    68    69    72
         !   73
G        !   39    86    87
H        !   41    88
I        !   25    27    28    29    62    63
         !   65    66    67    68    69    70
         !   72    73    74
J        !   95
M        !   58    68    78    79    81    82
         !   83    87    88
M1       !   43    90
M2       !   44    91    94    95
M3       !   45    92    95
M4       !   46    93    94
MT       !   49    79
N        !   22    26    27    62
TE       !   40    59    70    71    86
V1       !   58    64    78    81    82    83
V2       !   58    65    81    82    83
V3       !   58    66    82    83
V4       !   58    67    83
V5       !   58    72    87
V6       !   58    73    88
V7       !   58    69    79
VT       !   63    64    65    66    67
XC       !   26    29    63    65    66    67
         !   69    70    72    73
```

momenti di primo, secondo, terzo e quarto ordine dei cinque numeri 2,3,7,8,10
X= (2+3+7+8+10) /5=6 è il momento primo o media aritmetica
X↑2=(2↑2+3↑2+7↑2+8↑2+10↑2) /5=45.2 è il momento di secondo ordine
X↑3=(2↑3+3↑3+7↑3+8↑3+10↑3) /5=378 è il momento di terzo ordine
X↑4=(2↑4+3↑4+7↑4+8↑4+10↑4) /5=3318.8 è il momento di quarto ordine
Per concludere questa prima parte riguardante la statistica si riporta a seguito il listato del programma "Media" che permette di calcolare il valore delle medie e dei momenti relativi ad una variabile stocastica (cioè casuale) x che assume i valori X1, X2, ..., XN con frequenza f1, f2, ..., fN.

```
10 REM ***************************
11 REM * ELEMENTI DI STATISTICA:*
12 REM *     MEDIE E MOMENTI     *
13 REM *           DI            *
14 REM *  GUARDIONE MARIANGELA   *
15 REM ***************************
16 REM *    MENU' PRINCIPALE     *
17 REM *   ED INPUT DEI DATI     *
18 REM ***************************
19 GOSUB 89:PRINT"[CLEAR][2 DOWN]
   [2 RIGHT]CALCOLO DELLE MEDIE E
    DEI MOMENTI"
20 PRINT"[5 RIGHT]DI UNA VARIABIL
   E STATISTICA[3 DOWN]"
21 PRINT"DEFINIRE IL NUMERO DI VA
   LORI PER LA"
22 INPUT "VARIABILE STATISTICA: "
   ;N
23 PRINT
24 PRINT"FORNIRE I VALORI E LE RE
   LATIVE FREQUENZE"
25 FOR I=1 TO 50:NEXTI
26 DIM X(N),F(N)
27 FOR I=1 TO N
28 INPUT "X=";X(I):PRINT"[UP][20
   RIGHT]";:INPUT "F=";F(I)
29 NEXTI:PRINT"[3 DOWN]"
30 REM ***************************
31 REM *  CHIAMATA SUBROUTINE    *
32 REM *        DI CALCOLO       *
33 REM ***************************
34 GOSUB  59
35 REM ***************************
36 REM * GESTIONE DEI RISULTATI*
37 REM ***************************
38 PRINT"[CLEAR][2 DOWN]MEDIA ARI
   TMETICA= ";A
39 PRINT"[DOWN]MEDIA GEOMETRICA=
```

```
   ":G:PRINT
40 IF TE=1 THEN PRINT"MEDIA ARMON
   ICA NON DEFINITA[DOWN]":GOTO 4
   2
41 PRINT"MEDIA ARMONICA= ";H:PRIN
   T
42 PRINT"MOMENTI:"
43 PRINT"           .M1= ";M1
44 PRINT"           .M2= ";M2
45 PRINT"           .M3= ";M3
46 PRINT"           .M4= ";M4
47 PRINT"[DOWN]COEF. D'APPIATTIME
   NTO= ";AP
48 PRINT"[DOWN]COEF. D'ASIMMETRIA
   = ";AS
49 PRINT"[DOWN]MEDIA QUÀDRATIÇA=
   ";MT
50 END
51 REM ************************
52 REM *  SUBROUTINE CALCOLO   *
53 REM *  DELLE MEDIE E DEI    *
54 REM *       MOMENTI         *
55 REM ************************
56 REM *   INIZIALIZZAZIONI    *
57 REM ************************
58 TE=0:V1=0:V2=0:V3=0:V4=0:V5=1:
   V6=0:V7=0:M=0
59 REM ************************
60 REM *  CALCOLO SOMMATORIE   *
61 REM ************************
62 FOR I=1 TO N
63 VT=F(I)*X(I)
64 V1=V1+VT
65 V2=V2+VT*X(I)
66 V3=V3+VT*X(I)↑2
67 V4=V4+VT*X(I)↑3
68 M=M+F(I)
69 V7=V7+F(I)*X(I)↑2
70 IF X(I)=0 THEN TE=1:GOTO 74
71 IF TE=1 THEN 74
72 V5=V5*X(I)↑F(I)
73 V6=V6+F(I)/X(I)
74 NEXTI
75 REM ************************
76 REM *   CALCOLO FUNZIONI    *
77 REM ************************
78 A=V1/M
79 MT=(V7/M)↑.5
80 M1=A
81 M2=(V2-2*A*V1+A*A*M)/M
82 M3=(V3-3*A*V2+3*A*A*V1-M*A↑3)/
   M
83 M4=(V4-4*A*V3+6*A*A*V2-4*V1*A↑
   3+M*A↑4)/M
84 AP=M4/M2/M2
85 AS=M3/(M2↑1.5)
86 IF TE=1 THEN G=0:RETURN
87 G=V5↑(1/M)
88 H=M/V6:RETURN
89 PRINT"[CLEAR]":PRINT"[5 DOWN][
   6 RIGHT]%%%%%%%%%%%%%%%%%%%%%%
   %%%%%%"
90 PRINT"[6 RIGHT]%[RVS] ELEMENTI
    DI STATISTICA: [RVOFF]%"
91 PRINT"[6 RIGHT]%[RVS]      MEDI
   E E MOMENTI      [RVOFF]%
92 PRINT"[6 RIGHT]%[RVS]
     DI             [RVOFF]%"
93 PRINT"[6 RIGHT]%[RVS]  GUARDIO
   NE MARIANGELA   [RVOFF]%"
94 PRINT"[6 RIGHT]%%%%%%%%%%%%%%
   %%%%%%%%%%%%%%"
95 FOR J=1 TO 5000:NEXTJ:RETURN
```

LA N PRECISIONE

di Eugenio Coppari

(prima parte)

La matematica è al tempo stesso figlia e madre della moderna informatica. Una asserzione di questo genere è contraddittoria solamente in apparenza, poiché i legami che legano vicendevolmente le due discipline sono difficilmente separabili.

Questo breve commento iniziale è senza dubbio utile per introdurre l'argomento di questo articolo: il calcolo numerico. La prima parte della trattazione sarà di carattere essenzialmente teorico, mentre nella seconda il problema verrà analizzato soprattutto in riferimento alla precisione numerica che è consentita sulla maggiore parte degli home-computer. Questo fatto richiederà una attenta riflessione sull'algoritmc che è alla base del programma allegato all'articolo e che consente di trattare valori numerici superiori a quelli tradizionalmente accettati sui calcolatori appartenenti a questa fascia di mercato. Le operazioni consentite tra questi valori sono le 4 tradizionali: addizione, sottrazione, moltiplicazione e divisione.

Nel campo della teoria dei numeri, il calcolatore consente di esplorare valori assai più grandi rispetto a quelli che erano raggiungibili nei secoli passati. Quei procedimenti, essendo essenzialmente manuali, pagavano inevitabilmente il prezzo di una eccessiva lentezza e di una scarsa precisione.

Il calcolatore ha senza dubbio consentito, in questa specifica branca della matematica, una benefica iniezione di "giovinezza", dischiudendo nuovi orizzonti alla ricerca nel campo della teoria dei numeri. Non è azzardato, in questo caso, paragonare questa evoluzione con quella avvenuta in taluni campi della fisica. Scrutare il cielo, come d'altronde sarà possibile nei prossimi anni, attraverso un telescopio orbitante intorno alla Terra, rappresenterà un innegabile progresso rispetto alle nostre possibilità attuali di conoscenza dell'universo.

Le prime scoperte, che destarono notevole sensazione, furono quelle di Lander e Parkin che determinarono, mediante il computer, una soluzione della equazione X elevato alla quinta più Y elevato alla quinta più Z elevato alla quinta più U elevato alla quinta uguale a V elevato alla quinta potenza.

Il risultato è dato rispettivamente da questi valori:

$$X=27,\ Y=84,\ Z=110,$$
$$U=133,\ V=144$$

Si dimostrava quindi infondata la congettura di Eulero che riteneva priva di soluzioni questo genere di equazione. Il calcolo di pi greco, argomento che sarà oggetto di analisi su uno dei prossimi articoli della nostra rivista, è ormai giunto alla milionesima cifra decimale. Ricordiamo, al lettore che non lo rammentasse, che pi greco è dato dal rapporto tra la circonferenza e il suo diametro (3,14... etc.).

Ricerche di questo genere sono finalizzate alla scoperta di una periodicità nella disposizione dei decimali che compongono questo numero così misterioso e affascinante.

Nei nostri piccoli calcolatori domestici siamo abituati a chiamarle ugualmente con il nome di bit. Il tempo impiegato da un grande calcolatore per una operazione logica elementare (aperto o chiuso) è attualmente pari a 10 elevato alla meno 9 secondi: circa un miliardesimo di secondo. Esistono naturalmente dei limiti dettati dalla fisica per quanto concerne le massime dimensioni raggiungibili da un computer e la sua maggiore velocità di calcolo. Un calcolatore ottimale potrebbe essere costituito al più da un numero di elementi fondamentali pari a 10 elevato alla 126, aventi ognuno dimensioni non più piccole di quelle di un protone.

Inoltre questo calcolatore non potrà essere più grande dell'universo da noi conosciuto che, stando agli studi attuali, ha un diametro di 100 miliardi di anni luce. In base agli accertamenti teorici effettuati, ogni operazione logica elementare (aperto o chiuso) non potrà mai essere effettuata in un tempo inferiore a 3 moltiplicato per 10 elevato alla menc 24 secondi.

Questo è in effetti il tempo impiegato, in condizioni normali, dalla luce per attraversare il diametro di un protone. Anche se senza dubbio le capacità tecniche di un tale calcolatore teorico ci atteriscono, sarà importante rendersi conto di un fatto assai rilevante: continueranno ad esistere problemi teoricamente risolubili, cioè esprimibili sotto forma di un algoritmo ben definito, ma che nonostante ciò non saranno risolvibili a causa dei limiti fisici dell'universo in cui attualmente viviamo. Naturalmente uno dei motivi principali che ci conducono

ad una affermazione del genere, è il fatto che la maggiore velocità consentita in natura è quella della luce, pari a poco meno di 300.000 km al secondo. Quindi alcuni problemi di calcolo matematico, risolvibili in linea di principio, eccedono per quanto concerne la loro soluzione la capacità del più potente calcolatore ipotizzabile in natura.

Diamo ora alcuni esempi di quanto è stato affermato. Il gioco degli scacchi richiederebbe un tempo di esame di tutte le possibili situazioni di gioco, utilizzando il calcolatore ideale descritto in precedenza, pari a 20 miliardi di anni. Ritengo che assai difficilmente qualcuno di noi sarebbe ben disposto psicologicamente ad aspettare per un tale lasso di tempo!

La strategia degli scacchi si può studiare mediante una struttura ad albero di ricerca che naturalmente darà luogo ad una serie pressoché infinita di ramificazioni.

Anche altri "passatempi" possonc rientrare in questa non ristretta cerchia di problemi: ad esempio il gioco del Nim oppure le torri di Hanoi, possono ricorrere a un numero illimitato di fiammiferi.

Questa digressione, piuttosto lunga, compiuta in merito al problema dei limiti teorici di un calcolatore, ha come obiettivo quello di far comprendere i problemi che può suscitare l'utilizzo del calcolo al di fuori delle piccole quantità numeriche che usiamo giornalmente.

Difficilmente riusciamo a renderci conto del problema fisico, nonché matematico-filosofico, che implica condurre ai limiti estremi una abitudine quotidiana come quella del calcolo.

Sempre continuando nel nostro viaggio, che esplora i rapporti tra computer e calcolo numerico, ricordiamo il concetto di fattoriale: il prodotto dei primi N numeri naturali si chiama fattoriale di N o N fattoriale.

È giunta di recente la notizia che un gruppo di matematici giapponesi è riuscito a calcolare 15 milioni fattoriale! Il risultato occupa un blocco di carta di formato 132 colonne con una altezza pari a 25 centimetri: un risultato di questa rilevanza sarebbe stato improponibile forse anche solo alcuni mesi orsono.

Alcuni ricercatori di una università statunitense hanno scoperto, mediante circa 30 ore di lavoro in linguaggio macchina, che un numero di 77 cifre erroneamente ritenuto primo nel passato è in verità scomponibile in prodotto di fattori.

Lavori teorici di tale portata possono consentire applicazioni pratiche di notevole interesse. Si valuti ad esempio il problema della riservatezza delle informazioni, un quesito che necessita di un'impellente soluzione, data la sempre maggiore diffusione delle reti informatiche nei maggiori paesi del mondo. A questo proposito alcuni ricercatori hanno messo a punto un sistema crittografico a chiave pubblica basato sulla difficoltà di fattorizzazione di un numero molto grande, che sia per esempio, il prodotto di 2 numeri primi di 120 cifre. Naturalmente sarà statisticamente assai difficile, per una persona comune, trovare due fattori di questo genere, anche se l'avanzamento rapidissimo dei test di primalità, cioè di quelle prove che consentono di appurare se un numero è primo, può indurre ad avere seri dubbi sulla futura efficacia dei sistemi crittografici.

Gli elementi citati dovrebbero indurci a modificare sostanzialmente la nostra idea della ricerca matematica.

Il metodo moderno della ricerca scientifica ha due aspetti fondamentali: induttivo e deduttivo.

L'aspetto induttivo riguarda la ricerca di fatti o principi tramite osservazioni o esperimenti. L'aspetto deduttivo coinvolge ragionamenti logici che portano da proposizioni e principi assunti come premesse (assiomi o teoremi) a nuove proposizioni e principi.

A partire dalla scuola Galileiana i due aspetti hanno iniziato a fondersi in quasi tutte le scienze. E quindi anche nella matematica, con l'avvento dei calcolatori, non è più solamente una scienza prettamente deduttiva. In effetti il computer ha reso possibile, anche per questa disciplina, l'uso del termine "sperimentare". Tutti i problemi che abbiamo analizzato sino a questo momento richiedono, per quanto concerne la loro soluzione, velocità di calcolo difficilmente immaginabili dal possessore di un semplice home-computer.

Nei grandi calcolatori l'unità logica fondamentale viene chiamata porta, essa può ricevere due diversi segnali elettrici: aperto o chiuso (questa è la notazione maggiormente usata per rappresentare la logica a 2 stati tipica di ogni calcolatore). Queste porte sono costituite da componenti elettronici fondamentali e sono presenti all'interno di un calcolatore in un numero ben superiore a parecchie migliaia di elementi.

Il programma

Nella seconda parte dell'articolo viene esaminato il problema del calcolo su un home computer. Il programma commentato è scritto interamente in BASIC e quindi aldilà della memoria RAM richiesta per l'esecuzione del Software, non dovrebbero sussistere problemi di compatibilità tra i vari modelli COMMODORE, pur avendo io usato un 64. Il programma consente di utilizzare le 4 operazioni razionali (addizione, sottrazione, moltiplicazione, divisione) con numeri aventi al più 100 cifre.

Questo software risulterà particolarmente utile per coloro che richiedono maggiore precisione nell'ambito dei calcoli.

Sul COMMODORE 64 le costanti reali vengono visualizzate sino alla nona cifra. I numeri più piccoli di 0,01 e più grandi di 999999999 sono stampati in notazione scientifica. Una costante reale in notazione scientifica è composta da tre parti:

- mantissa
- lettera
- esponente.

La mantissa è un numero reale, e la lettera E viene usata per indicare che il numero viene rappresentato in formato esponenziale. In termini molto semplici. E vuol dire: per 10. C'è un limite per quanto concerne la grandezza dei numeri che può trattare il COMMODORE 64, infatti questi sono compresi tra 2,93873588 E-39 e 1,70141183 E+33.

Nelle righe da *1420* sino a *1480* viene creata la mascherina grafica che farà da contorno alla visualizzazione dei calcoli. Nelle righe da *1490* sino a *1525*, il programma vaglia quale tra le 4 operazioni disponibili è stata scelta per il calcolo. Nelle righe da *1530* sino a *2080* viene effettuata l'addizione: esaminiamone l'algoritmo.

Innanzitutto il programma vaglia se uno dei 2 addendi forniti contiene la virgola, procedendo in caso affermativo al loro incolonnamento. Questo è possibile post ponendo degli zeri ad uno dei due addendi. Si procede quindi all'addizione che dovrà tenere conto, naturalmente, di eventuali carry (riporti). Essa viene effettuata nelle righe che vanno da *1880* sino a *2080*. I due addendi vengono divisi e addizionati a gruppi di 8 cifre, in base al loro incolonnamento. Ognuna di queste piccole addizioni potrà provocare un riporto sull'addizione del gruppo successivo se si eccederà sino alla nona cifra per quanto concerne il risultato.

La routine della sottrazione è situata nelle righe che vanno da *2370* sino a *2650*. Anche nel caso della sottrazione, come già in quello della addizione, si deve procedere all'incolonnamento degli

addendi in caso di presenza della virgola.

Tra *2440* e *2650* viene effettuata la routine di sottrazione, come già nel caso della addizione gli elementi vengono sottratti a gruppi di 8, tenendo inoltre conto degli eventuali carry che si possono verificare. Nelle righe da *2740* sino a *2880* è situata la routine di input controllato, essa non consente l'entrata di elementi estranei al sistema. Ricordo a questo proposito che volendo digitare lo zero come prima cifra è necessario premere il carattere "." poiché, anche in questo caso, viene eseguita la normale rappresentazione adottata da tutti i modelli COMMODORE.

```
PROGRAMMA : NPREC PRG

-------*------------------ - - -
VAR.   ! LINEA DEL PROGRAMMA
-------*------------------ - - -
       !
A      !  1720  1730  1740  2260  2270  2280
       !  2290  2890  2910
A$     !  1580  1680  1700  1720  1770  1890
       !  2040  2070  2140  2240  2254  2260
       !  2320  2450  2610  2640  2900  2910
A1     !  1730  1740  1750  1880  1890  1900
       !  1920  2040  2070  2280  2290  2300
       !  2440  2450  2460  2480  2610  2640
A2     !  1740  1760  2290  2310
A3     !  3070  3080  3200  3210
A4     !  3080  3090
A7     !  3060  3070  3080
A7$    !  2990  3000  3055  3060  3100  3210
AQ     !  1700  1710  1980  2980  3055
AX     !  3400
B      !  1720  1750  2260  2270  2300  2910
B$     !  1600  1670  1700  1710  1720  1800
       !  1900  2160  2230  2254  2260  2350
       !  2460  2900  2910
B2     !  1750  1790  2300  2340
B7     !  3060
B7$    !  3020  3030  3055  3060  3190  3220
C      !  1720  1760  1780  1790  1810  2260
       !  2310  2330  2340  2360  2910
C$     !  1720  2260  2910
C1$    !  2460
C7     !  3060
C7$    !  3060
CF     !  1700  1710  3320  3340
CH$    !  2450  2480
CL$(   !  1700  2980  3300
CU     !  3090  3110
D      !  1880  1890  1900  1920  1950  2040
       !  2070  2440  2450  2460  2480  2520
       !  2610  2640
D$     !  1480  1490  1500  1510  1520
D1     !  3190  3220  3310
D8     !  3200  3210  3290
E      !  2740  2800  2835
E9     !  3230  3240  3450
EF$    !  1710
EV     !  2254  2545  2547  3055  3425
EW     !  2890  2900  2910  2920
F      !  1890  1910  2450  2470  2480  2490
F7     !  3210  3230
FI$    !  1710
G      !  1900  1910  2460  2470  2480  2490
G7     !  3220  3230
H2$    !  2254  3055
IN$    !  1580  1600  2140  2160  2740  2800
       !  2830  2860  2990  3020
JK     !  2740  2790  2870
KK$    !  8060
L$     !  2254
PE     !  1410  1470
R$     !  1700  1980  3400  3405  3410  3420
       !  3430
R1$    !  2545  2547
RH     !  2260  2545  2550  2910
RO     !  3000  3030  3400
S      !  1930  2060  2500  2630
SD     !  1670  1680  1690  1700  1960  2230
       !  2240  2250  2254  2530
SQ     !  3050  3405  3410
TI     !  2740  2760
TJ     !  3270
U      !  1910  1920  2470  2480  2490
U$     !  1920  1930  1940  2030  2470  2480
       !  2490  2500  2510  2600
U7     !  3230  3250
U7$    !  3250  3260  3270  3280  3440  3450
UZ     !  1920  2050  2060  2620  2630
V      !  2040  2050  2060  2610  2630
V$     !  2050  2060  2070  2620  2630  2640
V7     !  3010  3050
VW     !  1590  1670  1680  1690  2150  2230
       !  2240  2250
W2     !  1610  1670  1680  2170  2230  2240
W7     !  3040  3050
X$     !  1940  1960  1970  1980  2030  2510
       !  2530  2540  2545  2547  2560  2600
       !  2920  8050
X7$    !  3280  3300  3440
YX     !  1710
YY     !  1670  1680  2230  2240
Z      !  2800  2805  2810  2850  2860  2870
Z$     !  2750  2780  2790  2800  2830  2840
Z2$    !  3055
Z4     !  1590  1610  2150  2170  2740  2850
       !  2870  3010  3040
ZC     !  2740  2760
ZL     !  1590  1610  2150  2170  2740  2780
       !  2800  2805  2820  2835  2850  2870
       !  3010  3040
ZT     !  2740  2760
ZZ     !  3300  3320
```

Per porre termine all'ingresso dei dati relativi ad un determinato addendo o fattore ricordarsi di premere RETURN.

Nelle righe da *2980* sino a *3460* viene analizzata la moltiplicazione. Dopo la fase iniziale di input dei 2 fattori, si procede ad una loro prima valutazione nelle righe comprese tra *2980* e *3110*. La fase moltiplicativa avviene nelle righe comprese tra *3190* e *3340*, dove il primo fattore viene diviso in gruppi di 8 elementi ciascuno. Effettuata questa prima operazione, ogni cifra del secondo fattore, a partire dall'ultima, viene moltiplicata per ogni gruppo precedente. Nel caso in cui uno di questi prodotti fornisse un risultato di 9 cifre, si verificherebbe un riporto sulla moltiplicazione successiva.

Terminate tutte le moltiplicazioni, si procede alla fase di addizione di tutti i singoli risultati, dove provvederà il programma ad incolonnare esattamente tutti gli addendi.

Dovrebbe risultare chiaro che il prodotto è stato eseguito nella maniera che utilizziamo usualmente quando siamo provvisti di carta e matita.

La divisione e la parte di programma che la riguarda, meritano senza dubbio un discorso maggiormente particolareggiato. Quindi vi rimandiamo alla prossima puntata dell'"n precisione" dove il discorso verrà terminato definitivamente. Risulteranno inoltre particolarmente gradite altre routine dei lettori inerenti al tema del calcolo.

```
1000 REM ************************
1010 REM *                      *
1020 REM *     N  PRECISIONE    *
1030 REM *                      *
1040 REM *  CON LE OPERAZIONI : *
1050 REM *                      *
1060 REM *   +    -    *    /   *
1070 REM *                      *
1080 REM ************************
1090 REM *                      *
1100 REM *   AUTHOR SOFTWARE :  *
1110 REM *                      *
1120 REM *   EUGENIO COPPARI    *
1130 REM *                      *
1140 REM ************************
1150 REM *    VC 20           SI *
1160 REM *    COMMODORE 64    SI *
1170 REM *    COMMODORE 4000  SI *
1180 REM *    COMMODORE 8000  SI *
1190 REM *    COMMODORE 16    SI *
1220 REM *    COMMODORE PLUS  SI *
1230 REM ************************
1240 PRINT"[CLEAR]";
1250 REM
1260 REM ************************
1270 REM *                      *
1280 REM * SCELTA DELL'OPERAZIONE *
1290 REM *      DESIDERATA      *
1300 REM *                      *
1310 REM ************************
```

```
1320 POKE 53280,6:POKE 53281,6:REM
      BATTERE QUESTA RIGA SOLO PER
      IL COMMODORE 64
1330 PRINT TAB(23)"╭──────────────╮
      "
1340 PRINT TAB(23)"| N PRECISIONE |
      "
1350 PRINT TAB(23)"╰──────────────╯
      "
1370 PRINT"[3 UP]╭──────────────╮"
1380 PRINT"| +     -     *   |"
1390 PRINT"╰──────────────────╯"
1400 PRINT:PRINT
1410 FOR PE=1 TO 2
1420 PRINT"╭──────────────────────────
      ───────────────╮"
1430 PRINT"[UP]|
                       |"
1440 PRINT"[UP]|
                       |"
1450 PRINT"[UP]|
                       |"
1460 PRINT"[UP]╰──────────────────────
      ───────────────╯"
1470 PRINT"[UP]";:NEXTPE
1480 GET D$:IF D$="" THEN 1480
1490 IF D$<>"+" AND D$<>"-" AND D$<
      >"*" THEN 1480
1500 IF D$="+" THEN PRINT"[HOME]";
      TAB(59);"[RVS]"D$"[RVOFF]";SPC
      (182);:GOTO  1580
1510 IF D$="-" THEN PRINT"[HOME]";
      TAB(59);"[RVS]"D$"[RVOFF]";SPC
      (182);:GOTO  2090
1520 IF D$="*" THEN PRINT"[HOME]";
      TAB(59);"[RVS]"D$"[RVOFF]";SPC
      (182);:GOTO  2980
1530 REM *************************
1540 REM *                       *
1550 REM *       ADDIZIONE       *
1560 REM *                       *
1570 REM *************************
1580 GOSUB 2740:PRINT"[2 RIGHT]";:A
      $=IN$
1590 VW=ZL-Z4
1600 GOSUB 2740:PRINT"[2 RIGHT]";:B
      $=IN$
1610 W2=ZL-Z4
1620 REM *************************
1630 REM * EVENTUALE INCOLONNAMEN-*
1640 REM * DI ELEMENTI CON LA VIR-*
1650 REM *          GOLA         *
1660 REM *************************
1670 IF VW>W2 THEN FOR YY=1 TO (VW-
      W2):B$=B$+"0":NEXTYY:SD=VW:GOT
      O 1720
1680 IF W2>VW THEN FOR YY=1 TO (W2-
      VW):A$=A$+"0":NEXTYY:SD=W2:GOT
      O 1720
1690 SD=VW
1700 IF AQ=1 THEN A$=R$:B$=CL$(CF):
      SD=0
1710 IF AQ=1 AND CF<>1 THEN FOR YX=
      1 TO (CF-1):FI$=FI$+"0":NEXTYX
      :EF$=FI$:B$=B$+FI$:FI$=""
1720 A=LEN(A$):B=LEN(B$):IF B>A THE
      N C$=B$:B$=A$:A$=C$:C=B:B=A:A=
      C
1730 A1=INT(A/8)
1740 A2=(A1+1)*8-A
1750 B2=(A1+1)*8-B
1760 FOR C=1 TO A2
1770 A$="0"+A$
1780 NEXTC
1790 FOR C=1 TO B2
1800 B$="0"+B$
1810 NEXTC
1820 REM *************************
1830 REM * ADDIZIONE DEI DUE  AD- *
1840 REM * DENDI TENENDO CONTO DI *
1850 REM * EVENTUALI CARRY :      *
1870 REM *************************
1880 FOR D=1 TO (A1+1)
1890 F=VAL(MID$(A$,(A1+1-D)*8+1,8))
1900 G=VAL(MID$(B$,(A1+1-D)*8+1,8))
1910 U=F+G
1920 U$=MID$(STR$(U),2):IF (LEN(U$)
      =9 OR UZ<>0) AND (D<A1+1) THEN
       2030
1930 IF LEN(U$)<8 THEN FOR S=LEN(U$
      ) TO 7:U$="0"+U$:NEXTS
1940 X$=U$+X$
1950 NEXTD
1960 IF SD<>0 THEN X$=LEFT$(X$,LEN(
      X$)-SD)+"."+RIGHT$(X$,SD)
1970 IF MID$(X$,1,1)="0" THEN X$=MI
      D$(X$,2):GOTO 1970
1980 IF AQ=1 THEN R$=X$:X$="":GOTO
      3340
1990 GOTO 8050
2000 REM *************************
2010 REM *         CARRY :       *
2020 REM *************************
2030 X$=RIGHT$(U$,8)+X$
2040 V=VAL(MID$(A$,(A1-D)*8+1,8))+1
```

```
2050 IF V=100000000 THEN V$="000000
     00":UZ=1:GOTO  2070
2060 UZ=0:V$=MID$(STR$(V),2):IF LEN
     (V$)<8 THEN FOR S=LEN(V$) TO 7
     :V$="0"+V$:NEXT
2070 A$=LEFT$(A$,(A1-D)*8)+V$+RIGHT
     $(A$,D*8)
2080 GOTO 1950
2090 REM **************************
2100 REM *                        *
2110 REM *     SOTTRAZIONE    :   *
2120 REM *                        *
2130 REM **************************
2140 GOSUB 2740:PRINT"[2 RIGHT]";:A
     $=IN$
2150 VW=ZL-Z4
2160 GOSUB 2740:PRINT"[2 RIGHT]";:B
     $=IN$
2170 W2=ZL-Z4
2180 REM **************************
2190 REM * EVENTUALE INCOLONNAMEN-*
2200 REM * DI ELEMENTI CON LA VIR-*
2210 REM *        GOLA            *
2220 REM **************************
2230 IF VW>W2 THEN FOR YY=1 TO (VW-
     W2):B$=B$+"0":NEXTYY:SD=VW:GOT
     O 2260
2240 IF W2>VW THEN FOR YY=1 TO (W2-
     VW):A$=A$+"0":NEXTYY:SD=W2:GOT
     O 2260
2250 SD=VW
2251 REM  ROUTINE DIVISIONE
2254 IF EV=1 THEN A$=L$:B$=H2$:SD=0
     :
2260 A=LEN(A$):B=LEN(B$):IF B>A THE
     N C$=B$:B$=A$:A$=C$:C=B:B=A:A=
     C:RH=1
2270 IF B=A THEN 2890
2280 A1=INT(A/8)
2290 A2=(A1+1)*8-A
2300 B2=(A1+1)*8-B
2310 FOR C=1 TO A2
2320 A$="0"+A$
2330 NEXTC
2340 FOR C=1 TO B2
2350 B$="0"+B$
2360 NEXTC
2370 REM **************************
2380 REM *  SOTTRAZIONE   TRA  2  *
2390 REM *        ELEMENTI        *
2400 REM *     TENENDO CONTO      *
2410 REM *  DI EVENTUALI CARRY :  *
2430 REM **************************
2440 FOR D=1 TO (A1+1)
2450 CH$=MID$(A$,(A1+1-D)*8+1,8):F=
     VAL(CH$)
2460 C1$=MID$(B$,(A1+1-D)*8+1,8):G=
     VAL(C1$)
2470 IF F>=G THEN U=F-G:U$=MID$(STR
     $(U),2):GOTO 2500
2480 IF (G>F AND D<A1+1) THEN CH$="
     1"+CH$:F=VAL(CH$):U=F-G:U$=MID
     $(STR$(U),2):GOTO 2600
2490 IF G>F THEN U=F-G:U$=MID$(STR$
     (U),2)
2500 IF LEN(U$)<8 THEN FOR S=LEN(U$
     ) TO 7:U$="0"+U$:NEXTS
2510 X$=U$+X$
2520 NEXTD
2530 IF SD<>0 THEN X$=LEFT$(X$,LEN(
     X$)-SD)+"."+RIGHT$(X$,SD)
2540 IF MID$(X$,1,1)="0" THEN X$=MI
     D$(X$,2):GOTO 2540
2545 IF EV=1 AND RH=0 THEN R1$=X$:R
     ETURN
2547 IF EV=1 THEN R1$="-"+X$:RETURN
2550 IF RH=0 THEN 8050
2560 X$="-"+X$:GOTO 8050
2570 REM **************************
2580 REM *        CARRY           *
2590 REM **************************
2600 X$=RIGHT$(U$,8)+X$
2610 IF MID$(A$,(A1-D)*8+1,8)<>"000
     00000" THEN V=VAL(MID$(A$,(A1-
     D)*8+1,8))-1:GOTO 2630
2620 V$="99999999":UZ=1:GOTO 2640
2630 UZ=0:V$=MID$(STR$(V),2):IF LEN
     (V$)<8 THEN FOR S=LEN(V$) TO 7
     :V$="0"+V$:NEXTS
2640 A$=LEFT$(A$,(A1-D)*8)+V$+RIGHT
     $(A$,D*8)
2650 GOTO 2520
2660 REM **************************
2670 REM *    \ROUTINE DI INPUT    *
2680 REM *       CONTROLLATO      *
2690 REM * IMPEDISCE L'ACCESSO AD *
2700 REM * EVENTUALI ELEMENTI     *
2710 REM * ESTRANEI AL SISTEMA    *
2730 REM **************************
2740 IN$=" ":ZT=TI:ZC=2:Z4=0:ZL=0:J
     K=0:E=0
2750 GET Z$:IF  Z$<>"" THEN 2780
2760 IF ZT<=TI THEN PRINTMID$(" ※",
     ZC,1);"[LEFT]";ZC=3-ZC:ZT=TI+
     15
2770 GOTO 2750
```

```
2780 IF (Z$="0" AND ZL=0) THEN 2750
2790 IF JK=1 AND Z$="." THEN 2750
2800 Z=ASC(Z$):ZL=LEN(IN$):IF Z=46
     THEN E=E+1:GOTO 2835
2805 IF Z=13 AND ZL=1 THEN 20030
2810 IF Z<48 OR Z>57 THEN PRINT" [L
     EFT]";:GOTO 2850
2820 IF ZL>110 THEN 2750
2830 IN$=IN$+Z$
2835 IF ((ZL+E)/38)=INT((ZL+E)/38)
     THEN PRINT"[2 RIGHT]";
2840 PRINTZ$;
2850 IF Z=13 AND Z4=0 THEN Z4=ZL
2860 IF Z=13 THEN IN$=MID$(IN$,2):P
     RINT"[HOME]":PRINT"[8 DOWN]":R
     ETURN
2870 IF Z=46 THEN JK=1:Z4=ZL
2880 GOTO 2750
2890 FOR EW=1 TO A
2900 IF MID$(A$,EW,1)>MID$(B$,EW,1)
      THEN 2280
2910 IF MID$(B$,EW,1)>MID$(A$,EW,1)
      THEN C$=B$:B$=A$:A$=C$:C=B:B=
     A:A=C:RH=1:GOTO 2280
2920 NEXTEW:X$="0":GOTO 8050
2930 REM ***************************
2940 REM *                         *
2950 REM *        PRODOTTO    :    *
2960 REM *                         *
2970 REM ***************************
2980 A0=1:DIM CL$(200)
2990 GOSUB 2740:PRINT"[2 RIGHT]";:A
     7$=IN$
3000 IF MID$(A7$,1,1)="0" THEN A7$=
     MID$(A7$,2):RO=RO+1:GOTO 3000
3010 V7=ZL-Z4
3020 GOSUB 2740:PRINT"[2 RIGHT]";:B
     7$=IN$
3030 IF MID$(B7$,1,1)="0" THEN B7$=
     MID$(B7$,2):RO=RO+1:GOTO 3030
3040 PRINT:W7=ZL-Z4
3050 SQ=V7+W7
3055 IF EV=1 THEN A7$=H2$:B7$=Z2$:A
     0=1
3060 A7=LEN(A7$):B7=LEN(B7$):IF B7>
     A7 THEN C7$=B7$:B7$=A7$:A7$=C7
     $:C7=B7:B7=A7:A7=C7
3070 A3=INT(A7/8)
3080 A4=(A3+1)*8-A7
3090 FOR CU=1 TO A4
3100 A7$="0"+A7$
3110 NEXTCU
3120 REM ***************************
3130 REM * PRODOTTO DEI DUE  FAT-  *
3140 REM * TORI CON L'AUSILIO DEL  *
3150 REM * LA ROUTINE DI ADDIZIO-  *
3160 REM * NE PRECEDENTEMENTE      *
3170 REM * INTRODOTTA .            *
3180 REM ***************************
3190 FOR D1=LEN(B7$) TO 1 STEP -1
3200 FOR D8=1 TO (A3+1)
3210 F7=VAL(MID$(A7$,(A3+1-D8)*8+1,
     8))
3220 G7=VAL(MID$(B7$,D1,1))
3230 U7=F7*G7+E9
3240 E9=0
3250 U7$=MID$(STR$(U7),2)
3260 IF LEN(U7$)=9 THEN 3440
3270 IF LEN(U7$)<8 THEN FOR TJ=LEN(
     U7$) TO 7:U7$="0"+U7$:NEXTTJ
3280 X7$=U7$+X7$
3290 NEXTD8
3300 ZZ=ZZ+1:CL$(ZZ)=X7$:X7$=""
3310 NEXTD1
3320 FOR CF=1 TO ZZ
3330 GOTO 1700
3340 NEXTCF
3350 REM ***************************
3360 REM * POSIZIONAMENTO DELLA    *
3370 REM * VIRGOLA ALL'INTERNO     *
3380 REM * DEL RISULTATO FINALE    *
3390 REM ***************************
3400 IF RO<>0 THEN FOR AX=1 TO RO:R
     $="0"+R$:NEXTAX
3405 IF SQ>LEN(R$) THEN R$="."+R$:G
     OTO 3420
3410 R$=LEFT$(R$,LEN(R$)-SQ)+"."+RI
     GHT$(R$,SQ)
3420 IF MID$(R$,1,1)="0" THEN R$=MI
     D$(R$,2):GOTO 3420
3425 IF EV=1 THEN
3430 PRINT"[HOME][17 DOWN][2 RIGHT]
                 [RVS]IL RISULTATO E
     ' :[RVOFF]":PRINT"[DOWN]"R$:GO
     TO 20020
3440 X7$=MID$(U7$,2)+X7$
3450 E9=VAL(MID$(U7$,1,1))
3460 GOTO 3290
8000 REM ***************************
8010 REM *   ROUTINE DI FINE PRG   *
8020 REM ***************************
8050 PRINT"[HOME][17 DOWN][2 RIGHT]
                 [RVS]IL RISULTATO E'
     :[RVOFF]":PRINT"[DOWN]";X$
8060 GET KK$:IF KK$="" THEN 8060
8070 CLR :GOTO 1000
```

PROGRAMMIAMO I TASTI FUNZIONE

di Giancarlo de Cobelli

Molti possessori di Commodore 64 si saranno domandati come si utilizzano quei quattro tasti posti all'estrema destra della tastiera. La maggior parte dei giochi ed anche qualche programma di utilità fa uso di questi tasti per scegliere delle determinate funzioni che permettono di stabilire i parametri necessari al programma.
I tasti funzione numerati da F1 a F8 sono indirizzabili da programma BASIC per rendere più semplice l'utilizzo di un programma. Vengono testati dal programma utilizzando semplicemente una routine di GET, che a seconda di quale tasto funzione è stato premuto salterà ad una determinata subroutine che svolgerà la funzione prestabilità. Oppure si può attribuire ad ogni tasto funzione una determinata istruzione per poi rendere più semplice l'utilizzo di quest'ultima.
In totale possono essere utilizzati 16 tasti funzione usando la maniera diretta (f1, f3, f5, f7), con il tasto SHIFT (f2, f4, f6, f8), con il tasto COMMODORE (come nella maniera diretta) e con i tasti SHIFT più COMMODORE (come con SHIFT); si può utilizzare anche il tasto CTRL (Control).
I codici CHR$ dei tasti funzione si possono trovare su tutti i manuali che trattano la programmazione del Commodore 64.
Se si volesse utilizzare i suddetti tasti per memorizzare in ognuno di essi un token BASIC, per facilitare così la scrittura di un programma, bisogna per forza ricorrere al linguaggio macchina per evitare tutti quei problemi che il programma in BASIC comporterebbe.
Ad esempio se io digitassi le righe più avanti proposte avrei, ogni volta che premo il tasto f1, il caricamento da disco in memoria del primo file. Oppure, premendo f3 riporterei il fondo dello schermo al colore originale. Ma appena caricherò un nuovo file il programma in memoria verrà eliminato: perché il sistema operativo riaggiorna i puntatori di inizio programma, numero linea ed inizio variabili in funzione del file che verrà caricato.

```
10 GET A$:IF A$=" " THEN 10
20 IF A$=CHR$(133)
   THEN LOAD"*", 8
30 IF A$=CHR$(134)
   THEN POKE53280, 6
```

Il listato proposto è un programma

in BASIC. Contiene i dati in codice esadecimale del programma in linguaggio macchina che permette di attribuire a 16 tasti funzione una determinata stringa di caratteri sotto forma di DATA, a piacere dall'utente. Il programma è stato locato alla locazione decimale 53128 per permettere l'utilizzo anche di programmi molto lunghi, nonché del BASIC 4.0. Le linee da **128** a **130** contengono la solita routine di conversione da esadecimale in decimale per permettere al computer di interpretare i DATA che sono forniti sotto forma esa per renderne più facile la lettura e diminuire le possibilità di errore.

Dalla linea **143** inizia la routine che permette la lettura delle stringhe che attribuiscono a ciascun tasto la sua funzione. Si possono inserire stringhe non più lunghe di otto caratteri, in caso contrario avviene la segnalazione di errore e fine del programma; per personalizzare i DATA da me proposti basta sostituire al posto della mia stringa la stringa da voi desiderata facendo attenzione che l primo DATA di ogni riga non corrisponde al primo tasto funzione bensì all'ultimo.

Come spiegato nelle REM, per utilizzare i 16 tasti ridefiniti bisogna usare: la maniera diretta, premendo cioè il tasto da solo, shiftata, cioè il tasto funziona unitamente al tasto shift, contemporaneamente al tasto COMMODORE o il tasto CTRL (Control). In figura 2 c'è il disassemblato della routine in linguaggio macchina.

Questa routine fa uso dell'interrupt che significa richiesta di interruzione del programma. Infatti ogni sessantesimo di secondo il microprocessore interrompe il programma in corso per eseguire alcune routine di controllo. Fra questa routine salta ad eseguire un programma la cui locazione di partenza è contenuta nelle locazioni di memoria $0314 e $0315. Questo indirizzo è di solito $EA31 dove è locato un programma che svolge le normali routine di controllo del microprocessore.

Ora modificando questo indirizzo faremo eseguire al sistema operativo, prima del solito giro, una routine creata che inizierà all'indirizzo da noi modificato.

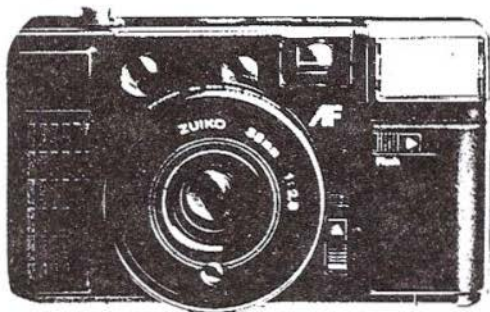
L'istruzione SEI disabilita la richiesta di interrupt per evitare che una successiva richiesta vada ad interrompere quella in corso. Poi avviene la modifica dell'indirizzo da cui dovrà iniziare la nostra routine ed infine viene riabilitato l'interrupt (CLI). A $CF9D inizia la nostra routine; essa salva i registri nello stack, controlla il tasto premuto e se quest'ultimo controllo riporterà un codice corrispondente a SHIFT o COMMODORE o CTRL testerà quale di essi è stato premuto per poi andare a leggere la stringa contenuta nel tasto funzione premuto. La lettura della stringa viene eseguita dalla routine che inizia a $CFE0. Dopo aver stampato sullo schermo il contenuto delle stringa od aver controllato che nessun tasto di quelli interessati è stato premuto rimetterà nei registri il loro contenuto originale e salterà alla normale routine di interuppt che come prima detto inizia a $EA31.

Questo comunque è solo un esempio di quello che si può fare utilizzando l'interrupt del processore 6502. Potrete così creare tutte le routine che da BASIC vi sembrano impossibili utilizzando questo semplicissimo sistema.

```
.,   CF88 78         SEI              .,   CFC4 EA         NOP
.,   CF89 A9 9C      LDA #$9C         .,   CFC5 EA         NOP
.,   CF8B 8D 14 03   STA $0314        .,   CFC6 A5 C5      LDA $C5
.,   CF8E A9 CF      LDA #$CF         .,   CFC8 A0 03      LDY #$03
.,   CF90 8D 15 03   STA $0315        .,   CFCA 84 FE      STY $FE
.,   CF93 58         CLI              .,   CFCC C5 FE      CMP $FE
.,   CF94 85 38      STA $38          .,   CFCE F0 11      BEQ $CFE1
.,   CF96 A9 00      LDA #$00         .,   CFD0 C8         INY
.,   CF98 85 37      STA $37          .,   CFD1 E8         INX
.,   CF9A 60         RTS              .,   CFD2 E8         INX
.,   CF9B EA         NOP              .,   CFD3 E8         INX
.,   CF9C EA         NOP              .,   CFD4 E8         INX
.,   CF9D 48         PHA              .,   CFD5 E8         INX
.,   CF9E 8A         TXA              .,   CFD6 E8         INX
.,   CF9F 48         PHA              .,   CFD7 E8         INX
.,   CFA0 98         TYA              .,   CFD8 E8         INX
.,   CFA1 48         PHA              .,   CFD9 C0 07      CPY #$07
.,   CFA2 A5 C5      LDA $C5          .,   CFDB D0 ED      BNE $CFCA
.,   CFA4 C5 FB      CMP $FB          .,   CFDD 4C F6 CF   JMP $CFF6
.,   CFA6 F0 4E      BEQ $CFF6        .,   CFE0 EA         NOP
.,   CFA8 85 FB      STA $FB          .,   CFE1 8A         TXA
.,   CFAA EA         NOP              .,   CFE2 85 FC      STA $FC
.,   CFAB A2 00      LDX #$00         .,   CFE4 A9 CF      LDA #$CF
.,   CFAD AD 8D 02   LDA $028D        .,   CFE6 85 FD      STA $FD
.,   CFB0 A8         TAY              .,   CFE8 A0 00      LDY #$00
.,   CFB1 C0 00      CPY #$00         .,   CFEA B1 FC      LDA ($FC),Y
.,   CFB3 F0 09      BEQ $CFBE        .,   CFEC 99 77 02   STA $0277,Y
.,   CFB5 8A         TXA              .,   CFEF C8         INY
.,   CFB6 18         CLC              .,   CFF0 C0 08      CPY #$08
.,   CFB7 69 20      ADC #$20         .,   CFF2 D0 F6      BNE $CFEA
.,   CFB9 AA         TAX              .,   CFF4 84 C6      STY $C6
.,   CFBA 88         DEY              .,   CFF6 68         PLA
.,   CFBB 18         CLC              .,   CFF7 A8         TAY
.,   CFBC 90 F3      BCC $CFB1        .,   CFF8 68         PLA
.,   CFBE E0 60      CPX #$60         .,   CFF9 AA         TAX
.,   CFC0 30 02      BMI $CFC4        .,   CFFA 68         PLA
.,   CFC2 A2 60      LDX #$60         .,   CFFB 4C 31 EA   JMP $EA31
                                      .,   CFFE 00         BRK
```

```
100 REM **************************
101 REM *                        *
102 REM *     TASTI FUNZIONE     *
103 REM *                        *
104 REM **************************
105 REM *                        *
106 REM * AUTHOR SOFTWARE :      *
107 REM *                        *
108 REM *  GIANCARLO DE COBELLI  *
109 REM *                        *
110 REM **************************
111 REM *                        *
112 REM * VIC 20 + EXP.       NO *
113 REM * COMMODORE 64        SI *
114 REM * COMMODORE 4000      NO *
115 REM * COMMODORE 8000      NO *
116 REM * COMMODORE 16        NO *
117 REM * COMMODORE PLUS 4    NO *
120 REM *                        *
121 REM **************************
122 :
123 FOR K=53128 TO 53247:READ A$
124 :
125 REM **************************
126 REM * CONVERSIONE HEX TO DEC *
127 REM **************************
128 X=ASC(LEFT$(A$,1))-48:IF X>9 T
    HEN X=X-7
129 Y=ASC(RIGHT$(A$,1))-48:IF Y>9
    THEN Y=Y-7
130 A=X*16+Y:POKE K,A:NEXT
131 :
132 DATA   78,A9,9C,8D,14,03,A9,CF
133 DATA   8D,15,03,58,85,38,A9,00
134 DATA   85,37,60,EA,EA,48,8A,48
135 DATA   98,48,A5,C5,C5,FB,F0,4E
136 DATA   85,FB,EA,A2,00,AD,8D,02
137 DATA   A8,C0,00,F0,09,8A,18,69
138 DATA   20,AA,88,18,90,F3,E0,60
139 DATA   30,02,A2,60,EA,EA,A5,C5
140 DATA   A0,03,84,FE,C5,FE,F0,11
141 DATA   C8,E8,E8,E8,E8,E8,E8,E8
142 DATA   E8,C0,07,D0,ED,4C,F6,CF
143 DATA   EA,8A,85,FC,A9,CF,85,FD
144 DATA   A0,00,B1,FC,99,77,02,C8
145 DATA   C0,08,D0,F6,84,C6,68,A8
146 DATA   68,AA,68,4C,31,EA,00,FF
147 :
148 REM **************************
149 REM *     LETTURA DATA       *
150 REM *TOK.BASIC=TASTO FUNZIONE*
151 REM **************************
152 :
153 FOR B=0 TO 15:READ B$
154 L=LEN(B$):IF L>8 THEN PRINTF;B
    $:PRINT"MAX. 8 CARATTERI":END
155 FOR I=1 TO 8
156 IF I>L THEN V=0:GOTO 158
157 V=ASC(MID$(B$,I,1)):IF V=95 TH
    EN V=13
158 POKE 52991+I+B*8,V:NEXTI,B
159 CLR
160 SYS53128
161 NEW
162 :
163 REM **************************
164 REM *   SENZA  TASTO SHIFT   *
165 REM *  F7    F1    F3    F5  *
166 REM **************************
167 :
168 DATA "PEEK","FOR","IF","GOTO"
169 :
170 REM **************************
171 REM *   CON    TASTO SHIFT   *
172 REM *  F8    F2    F4    F6  *
173 REM **************************
174 :
175 DATA "POKE","NEXT","THEN","GOS
    UB"
176 :
177 REM **************************
178 REM *  CON TASTO COMMODORE   *
179 REM *  F7    F1    F3    F5  *
180 REM **************************
181 :
182 DATA "CLOSE","OPEN","PRINT#","
    INPUT#"
183 :
184 REM **************************
185 REM *   CON TASTO  CONTROL   *
186 REM *  F7    F1    F3    F5  *
187 REM **************************
188 :
189 DATA "RUN←","LOAD","SAVE","LIS
    T"
```

L'ARCHIVIO FOTOGRAFICO

di **Renzo Arrigoni**

Questo programma è uno stringato "data base" dedicato alla fotografia. È stato fatto per aiutare quei fotoamatori che hanno accumulato centinaia e centinaia di fotografie e vogliono mettere ordine nel proprio archivio, di solito lo dico per esperienza alquanto caotica. Il programma consente di codificare le foto in modo razionale ed offre la possibilità di ricerche all'interno dell'archivio stesso secondo le specifiche desiderate.

Le fotografie vengono classificate con un numero progressivo, viene assegnato un titolo ed un codice particolare formato da sei numeri che servono ad individuarne le principali caratteristiche.
Naturalmente l'archivio viene memorizzato, su nastro o su disco, in files ai quali l'utente attribuirà nomi diversi (ad es. "stampe" o "diacolor 4" o altri) per individuare gruppi omogenei di fotografie.

Altre operazioni possibili sono poi quelle di stampa dell'archivio, di ricerca con stampa delle foto che si vogliono selezionare, variazione delle caratteristiche assegnate e interpretazione del codice delle foto.

Fra le caratteristiche di questo programma, la più interessante permette di personalizzare secondo le proprie esigenze il programma stesso, con modifiche che si effettuano con grande semplicità. In altre parole questo può considerarsi alla stregua di un canovaccio sul quale ciascuno, se lo desidera, può trovare il modo migliore di lavorare. Per questo il programma è stato strutturato a blocchi tra loro indipendenti, o comunque ben distinti, in modo da poter intervenire all'interno di essi senza problemi di funzionamento.
Passiamo ora ad analizzare il programma nel dettaglio per sottolineare quali siano i punti in cui è possibile o utile intervenire.

Il programma

Alla riga **10** viene dichiarata la dimensione del vettore a $ per la capienza in memoria di duemila fotografie. In realtà la memoria del calcolatore consentirebbe dimensionamenti più ampi. A questo proposito viene segnalata, nelle note al programma, come fare per cambiare il dimensionamento. 2000 elementi rappresentano una quantità prudenziale non essendo possibile determinare a priori la lunghezza del dato variabile cioè il titolo della fotografia. Tale titolo andrebbe contenuto mediamente, ma non obbligatoriamente, in venti/venticinque caratteri. Se la capienza di 2000 foto sembrasse modesta, si ovvierà creando, con files di nome diverso, archivi paralleli contenenti ognuno materiale omogeneo.

Dalla linea **200** alla **480** viene presentato il menù con tutte le opzioni possibili.
Vale la pena di ricordare, e questo vale anche per il seguito, che tutti i caratteri grafici per ottenere i numeri in revese, o quelli lampeggianti, possono essere tranquillamente eliminati a favore di un ulteriore risparmio di memoria.

Dalla **500** alla **599** si trovano le istruzioni "DATA" che serviranno a codificare le singole fotografie. Verranno presentati sei format (se ne possono creare altri), di dieci linee ciascuno, contenenti informazioni diverse.
Per ognuno di essi si segnalerà la scelta battendo il numero corrispondente.

Su questi si può intervenire a piacimento, sostituendo con altri quelli

```
C R O S S   R E F E R E N C E

PROGRAMMA : ARCHFOTO PRG

------*---------------- - - -
VAR.  ! LINEA DEL PROGRAMMA
------*---------------- - - -
      !
A$(   !  150  1130  1180  1240  1630  1640
      ! 1730  1740  2140  2150  2250  2255
      ! 2680  2720  3670  3675  3680  3720
      ! 3800  4150  4180  4220  4350  4520
B$(   ! 1180  1200  1240  4130  4135  4140
      ! 4160  4180
B(    ! 3060  3080  3090
BB$(  ! 3060
BL    ! 1126  4900
CD    ! 3020  3060
CF    ! 3020
CF$   ! 3010  3020  3040  3060
D$    ! 1160  3080  3090  4120
DD    ! 3040  3120
EM$   ! 1800  1830
EN    ! 1800  1810  1830
ES    ! 1800  1830
ET    ! 1800  1830
FD    ! 3040
FF    ! 3040
I     ! 1120  1130  1180  1240  1260  1630
      ! 1640  1650  1730  1740  1750  2140
      ! 2150  2160  2250  2255  2257  2680
      ! 2720  3660  3670  3675  3680  3720
      ! 3800  4150  4180  4220  4350  4510
      ! 4520
I$    ! 1130
J     ! 1160  3080  3090  4120
K     ! 1080  1140  1180  1200  1240  1260
      ! 4100  4130  4135  4140  4155  4160
      ! 4180  4200
K$    ! 1130
M     ! 4150  4155  4180
N     ! 1040  1120  1180  2580  2680  2920
      ! 3560  3660  4520
NF$   ! 1600  1620  1710  1720  2100  2120
      ! 2220  2240
Q     !  195
R     ! 4150  4180  4220
T     ! 1080  1120  1180  2580  2680  2940
      ! 3580  3660  3720  4520
TA    ! 3650  3660  3670
UN    ! 1040
X     !  450   460  3060  3080  3090
XX    ! 3760
XX$   ! 1540  1560  2040  2060  2560  2580
      ! 2600  2700  3200  4300  4350  5000
Z     !  440
Z$    !  400   420   440
```

ritenuti inutili o non essenziali, secondo il personale modo di vedere l'archivio. Per facilitare le operazioni di modifica sono stati scritti ognuno su una propria linea.
Il criterio da seguire è, come ovvio, quello di raggruppare dieci a dieci i dati tra loro omogenei, possibilmente dedicando quello di testa a titolo del gruppo (salvo intervenire nel programma per recuperare anche questo dato a possibile opzione).
Qualora invece non si desideri utilizzarli tutti, si potrà eliminare quelli inutili scrivendo: DATA" ".

Le righe **1000/1300** assegnano i valori agli elementi dell'archivio, numerandoli progressivamente. Vale quanto detto prima a proposito del titolo, che potrà essere anche uno spazio vuoto, qualora non si intenda assegnarlo.

Le successive linee, da **1500** a **1830,** riguardano la registrazione e la lettura dei files, da nastro o da disco. In fase di battitura potrà essere opportuno trascrivere le linee relative alla periferia che si intende usare. Così, chi possiede il solo registratore, eviterà di trascrivere le linee da **1700** a **1830** comprese, e da **2200** a **2260** comprese, correggendo eventualmente anche dove viene richiesta l'opzione Nastro—Disco.

Le righe che vanno da **3000** a **3220** servono per leggere in chiaro il codice assegnato alla fotografia. Si inserirà l'intera stringa relativa alla foto, ad esclusione del titolo; verrà letto il numero della fotografia e confermato il significato del codice.

La routine da **3500** a **3800** stampa il contenuto dell'archivio.
Utilizzando, per esempio, carta adesiva si potrebbe sveltire il lavoro di archiviazione. La routine di ricerca, da **4000** a **4380,** consente di visionare le foto in memoria ed estrarre dall'archivio quelle che hanno le caratteristiche desiderate.

La ricerca avviene sulla base di una o più scelte contemporanee (al limite tutte), e si esegue digitando asterisco nei format che non sono essenziali alla ricerca e battendo invece il numero di codice in quei format dove si desidera effettuare la selezione.

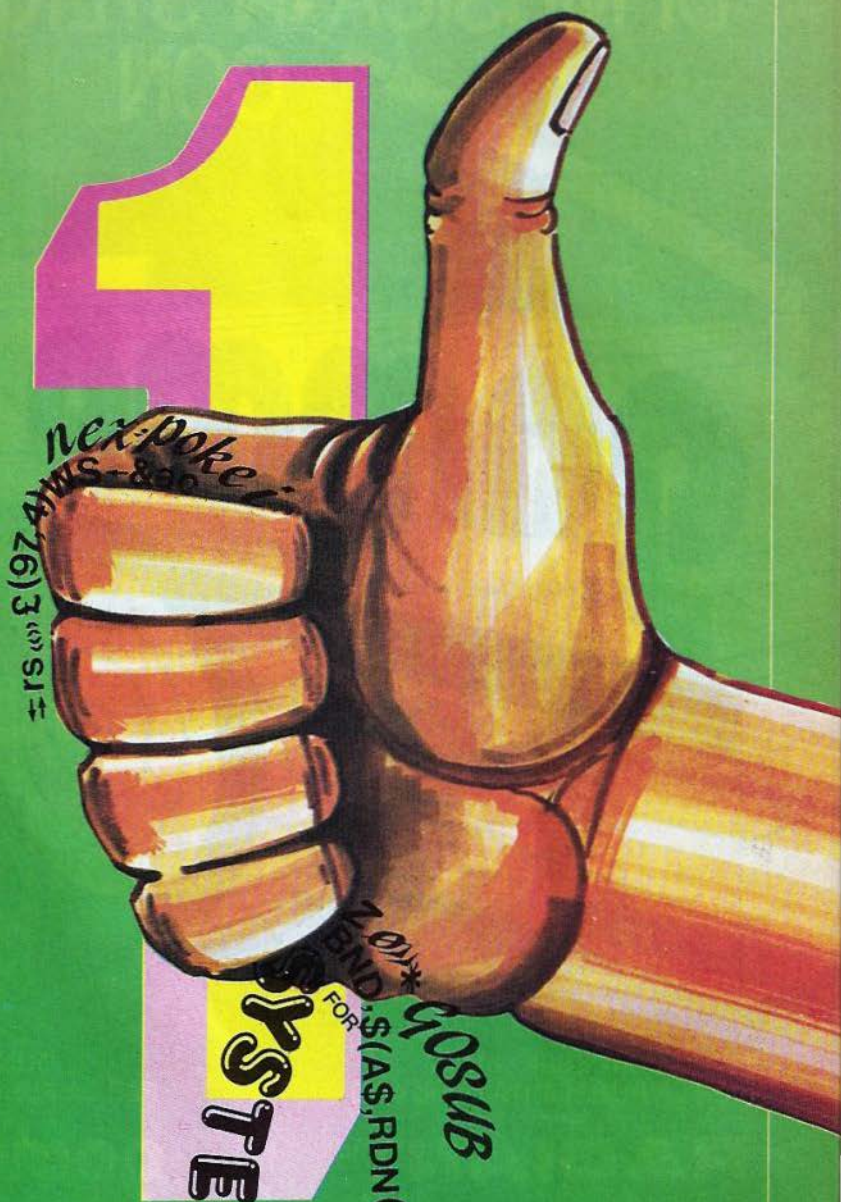
Le restanti istruzioni riguardano le modifiche che si rendessero necessarie, su tutti i campi previsti. Bisognerà quindi richiamare la fotografia col numero progressivo precedentemente assegnato e reinserire il titolo eventualmente cambiato e i restanti dati, come richiesto di volta in volta.

```
50 REM ***********************
51 REM *                     *
52 REM * L' ARCHIVIO         *
53 REM *       FOTOGRAFICO   *
54 REM *                     *
60 REM ***********************
61 REM *                     *
62 REM * AUTORI:             *
63 REM *                     *
64 REM *   MARCO  E  RENZO   *
65 REM *       ARRIGONI      *
66 REM *                     *
68 REM ***********************
70 REM * VIC 20           NO *
71 REM * VIC 20 + EXP     SI *
72 REM * COMMODORE 64     SI *
73 REM * SERIE 4000/8000 SI  *
74 REM * COMMODORE 16     SI *
75 REM * PLUS4            SI *
78 REM ***********************
110 PRINT"[CLEAR]"
120 PRINTCHR$(5):POKE 53281,5:POKE
    53280,5:REM  ABOLIRE SE SI US
    A TV BIANCO/NERO
140 PRINTCHR$(14)
145 REM **********************
146 REM * DIMENSIONE VETTORE *
147 REM **********************
150 DIM A$(2000):REM  SE VIENE AUM
    ENTATO IL DIMENSIONAMENTO, BIS
    OGNA CAMBIARE
151 REM  TUTTI I CICLI FOR I = 1 T
    O 2000
170 REM      A R C H I V I O    F O
     T O G R A F I C O
180 PRINT"[5 DOWN]":PRINT"
       ♠ _ ‾ | \ X \ Γ"
190 PRINT"[2 DOWN]":PRINT"
      _ Γ | Γ | _ ♠ _ \ ‾ Γ"
195 FOR O=1 TO 2000:NEXT
199 REM  MENU
200 PRINT"[CLEAR]":PRINT"[RVS]
               [RVOFF] \‾// [RVS]
                      [RVOFF]":PRI
    NT
220 PRINT"     - [RVS]1[RVOFF] - ⊣L
    ASSIFICAZIONE":PRINT
240 PRINT"     - [RVS]2[RVOFF] - \E
    MORIZZAZIONE FILE":PRINT
260 PRINT"     - [RVS]3[RVOFF] - LE
    TTURA FILE":PRINT
280 PRINT"     - [RVS]4[RVOFF] - ‾O
    NTENUTO ARCHIVIO IN MEMORIA":P
    RINT
300 PRINT"     - [RVS]5[RVOFF] - ‾E
    CODIFICA":PRINT
320 PRINT"     - [RVS]6[RVOFF] - ●T
    AMPA":PRINT
340 PRINT"     - [RVS]7[RVOFF] - _I
    CERCA":PRINT
350 PRINT"     - [RVS]8[RVOFF] - XA
    RIAZIONI":PRINT
360 PRINT
380 PRINT"        1 2 3 4 5 6 7 8 "
400 GET Z$:IF Z$="" THEN 450
420 IF Z$<"1" OR Z$>"8" THEN 450
440 Z=VAL(Z$):ON ZGOTO 1000,1500,2
    000,2500,3000,3500,4000,4500
450 FOR X=1 TO 300:NEXT
460 PRINT"[UP]        [RVS] 1 2 3 4
    5 6 7 8 [RVOFF]":FOR X=1 TO 30
    0:NEXT:PRINT"[2 UP]"
480 GOTO 380
500 DATA SOGGETTI - 9 - PER A|LTRI
501 DATA - 1 - FOTO D'ARCHITETTURA
502 DATA - 2 - PAESAGGI DI CAMPAGN
    A
503 DATA - 3 - PAESAGGI DI MARE
504 DATA - 4 - PAESAGGI DI MONTAGN
    A
505 DATA - 5 - CITTA'
506 DATA - 6 - PAESAGGI DIVERSI
507 DATA - 7 - REPORTAGE
508 DATA - 8 - RITRATTI
509 DATA ""
510 DATA SOGGETTI   - 9 - SE GIA'
    SCELTO
511 DATA - 1 - GLAMOUR
512 DATA - 2 - FOTO SPORTIVE
513 DATA - 3 - FOTO SUBACQUEE
514 DATA - 4 - ELABORAZIONI
515 DATA - 5 - MACRO FIORI
516 DATA - 6 - MACRO INSETTI
517 DATA - 7 - MACRO FUNGHI
518 DATA - 8 - DIVERSE
519 DATA ""
520 DATA GIUDIZIO
521 DATA - 1 - FOTO MEDIOCRE
522 DATA - 2 - FOTO DISCRETA
523 DATA - 3 - FOTO BUONA
524 DATA - 4 - FOTO OTTIMA
525 DATA - 5 - FOTO ECCEZIONALE
526 DATA - 6 - FOTO DA CONCORSO
527 DATA - 7 - FOTO GIA' PREMIATA
528 DATA - 8 - FOTO DA PRESENTARE
    A CONCORSI
```

```
529 DATA - 9 - SENZA GIUDIZIO
530 DATA CONDIZIONI DI LUCE
531 DATA - 1 - NATURALE ALL'ALBA
532 DATA - 2 - NATURALE AL TRAMONT
    O
533 DATA - 3 - MEZZOGIORNO
534 DATA - 4 - CONDIZIONI NORMALI
535 DATA - 5 - ARTIFICIALE DIFFUSA
536 DATA - 6 - ARTIFICIALE DIRETTA
537 DATA - 7 - LAMPADE AD INCANDES
    CENZA
538 DATA - 8 - NOTTURNO
539 DATA - 9 - NON CLASSIFICATA
540 DATA OBIETTIVO
541 DATA - 1 - GRANDANGOLO
542 DATA - 2 - GRANDANGOLO MEDIO
543 DATA - 3 - NORMALE
544 DATA - 4 - MEDIO TELE
545 DATA - 5 - LUNGA FOCALE
546 DATA - 6 - SOFFIETTO MACRO
547 DATA - 7 - LENTE 1 X
548 DATA - 8 - LENTE 2 X
549 DATA - 9 - LENTE 3 X
550 DATA .       TEMPI
     DIAFRAMMI
551 DATA - 1 - DA 1/125 A 1/2000
     APERTO
552 DATA - 2 - DA 1/125 A 1/2000
     MEDIO
553 DATA - 3 - DA 1/125 A 1/2000
     CHIUSO
554 DATA - 4 - DA 1/30  A 1/125
     APERTO
555 DATA - 5 - DA 1/30  A 1/125
     MEDIO
556 DATA - 6 - DA 1/30  A 1/125
     CHIUSO
557 DATA - 7 - DA SEC.  A 1/15
     APERTO
558 DATA - 8 - DA SEC.  A 1/15
     MEDIO
559 DATA - 9 - DA SEC.  A 1/15
     CHIUSO
996 REM *******************
997 REM * CLASSIFICAZIONE  *
998 REM * DELLA FOTOGRAFIA *
999 REM *******************
1000 PRINT"[CLEAR]"
1020 PRINT: PRINT"/UMERO DELL'ULTIM
     A FOTO CATALOGATA":PRINT
1040 INPUT UN:N=UN+1
1060 PRINT:PRINT"QUANTE FOTO PENSI
     DI CATALOGARE OGGI ?"
1070 PRINT:PRINT"     ( EVENTUALMENT
     E PER TERMINARE  "
1075 PRINT"     L'IMMISSIONE DEI DAT
     I, BATTERE * )":PRINT
1080 INPUT K:T=K-1
1100 PRINT"[CLSAR]"
1120 FOR I=N TO N+T:PRINT:PRINT
1125 PRINT"[RVS]          [RVOFF] IT
     OLO DELLA FOTOGRAFIA [RVS]
        [RVOFF]"
1126 GOSUB 4900:PRINT"[22 RIGHT]BYT
     E LIBERI "BL
1130 INPUT K$:I$=STR$(I):A$(I)=I$+"
      "+K$+" "+A$(I):PRINT"[CLEAR]"
     :PRINTA$(I)
1140 FOR K=0 TO 5
1160 FOR J=0 TO 9:REND D$:PRINT TAB
     (5)D$:PRINT:NEXTJ
1180 GET B$(K):IF B$(K)="*" THEN K=
     6:A$(I)="":I=N+T:GOTO 1260
1200 IF B$(K)<"1" OR B$(K)>"9" THEN
      1180
1220 PRINT"[CLEAR]"
1240 A$(I)=A$(I)+B$(K):PRINTA$(I)
1260 NEXTK:PRINT"[CLEAR]":RESTORE :
     NEXTI:RESTORE
1300 GOTO 200
1497 REM ******************
1498 REM * SCRITTURA FILE *
1499 REM ******************
1500 PRINT"[CLEAR]":PRINT"[RVS]
            [RVOFF] \¯\/¯-\****\/¯/¯
       [RVS]              [RVOFF]":PRI
     NT
1501 REM  MEMORIZZAZIONE
1520 PRINT"  SU   [RVS]/[RVOFF]ASTR
     O              SU   [RVS]¯[RV
     OFF]ISCO"
1540 GET XX$:IF XX$<>"N" AND XX$<>"
     D" THEN 1540
1560 IF XX$="D" THEN 1700
1580 PRINT"[3 DOWN]":PRINT"     *CRI
     TTURA SU NASTRO DEL FILE":PRIN
     T"[2 DOWN]"
1600 INPUT "     CHIAMATO";NF$
1620 OPEN 1,1,1,NF$
1630 FOR I=1 TO 2000:PRINT#1,A$(I)
1640 IF A$(I)="" THEN I=2000:PRINT#
     1,"***********"
1650 NEXTI
1680 CLOSE 1:GOTO 200
1700 PRINT"[3 DOWN]":PRINT"     *CRI
     TTURA SU DISCO DEL FILE":PRINT
```

```
     "[2 DOWN]"
1710 INPUT "     CHIAMATO";NF$
1715 OPEN 15,8,15:PRINT#15,"I"
1720 OPEN 2,8,2,"0:"+NF$+",S,W":GOS
     UB 1800
1730 FOR I=1 TO 2000:PRINT#2,A$(I)
1740 IF A$(I)="" THEN I=2000:PRINT#
     2,"***********"
1750 NEXTI
1760 CLOSE 2:CLOSE 15:GOTO 200
1800 INPUT#15,EN,EM$,ET,ES
1810 IF EN=0 THEN RETURN
1820 PRINT"ERRORE DISCO"
1830 PRINTEN,EM$,ET,ES:STOP
1997 REM ***************
1998 REM * LETTURA FILE *
1999 REM ***************
2000 PRINT"[CLEAR]":PRINT"[RVS]
              [RVOFF] L⌐!! ⁄—♠ —\L⌐
     [RVS]              [RVOFF]":PRI
     NT
2001 REM  LETTURA FILE
2020 PRINT"  DA   [RVS]/[RVOFF]ASTR
     O              DA   [RVS]⌐[RV
     OFF]ISCO"
2040 GET XX$:IF XX$<>"N" AND XX$<>"
     D" THEN 2040
2060 IF XX$="D" THEN 2200
2080 PRINT"[3 DOWN]":PRINT"     LETT
     URA DA NASTRO DEL FILE":PRINT"
     [2 DOWN]"
2100 INPUT "     CHIAMATO";NF$
2120 OPEN 1,1,0,NF$
2140 FOR I=1 TO 2000:INPUT#1,A$(I)
2150 IF A$(I)="***********" THEN A$
     (I)="":I=2000
2160 NEXTI
2180 CLOSE 1:GOTO 200
2200 PRINT"[3 DOWN]":PRINT"     LETT
     URA DA DISCO DEL FILE":PRINT"[
     2 DOWN]"
2220 INPUT "     CHIAMATO";NF$
2230 OPEN 15,8,15
2240 OPEN 2,8,2,"0:"+NF$+",S,R":GOS
     UB 1800
2250 FOR I=1 TO 2000:INPUT#2,A$(I)
2255 IF A$(I)="***********" THEN A$
     (I)="":I=2000
2257 NEXTI
2260 CLOSE 2:CLOSE 15:GOTO 200
2496 REM   *********************
2497 REM   * CONTENUTO VETTORE *
2498 REM   * IN  MEMORIA       *
2499 REM   *********************
2500 PRINT"[CLEAR]":PRINT
2505 PRINT"[RVS]          [RVOFF] ♠TAM
     PA CONTENUTO MEMORIA [RVS]
        [RVOFF]":PRINT
2520 PRINT" ♠TAMPA        [RVS]![RVOF
     F]OTALE      [RVS]⌐[RVOFF]ARZI
     ALE"
2530 PRINT:PRINT" !ATTERE - RETURN
     - PER SCROLLING"
2540 GOSUB 5000
2560 IF XX$<>"T" AND XX$<>"P" AND X
     X$<>CHR$(13) THEN 2540
2580 IF XX$="T" THEN N=1:T=2000:GOT
     O 2680
2600 IF XX$="P" OR XX$=CHR$(13) THE
     N GOSUB 2900:GOTO 2680
2620 GOTO 200
2680 FOR I=N TO T:PRINTA$(I)
2700 IF XX$=CHR$(13) THEN GOSUB 500
     0
2720 IF A$(I)="" THEN I=2000
2739 REM    BATTI SPAZIO PER MENU
2740 NEXT:PRINT"              !ATTI    ♠
     ⌐♠♠\⌐  PER \⌐/⁄"
2760 GOSUB 5000:GOTO 200
2900 PRINT:PRINT"  ♠TAMPA DA NUMERO
     ":PRINT
2920 INPUT N:PRINT
2940 PRINT"  A NUMERO":PRINT:INPUT
     T:RETURN
2997 REM   ******************
2998 REM   * DECODIFICA FOTO *
2999 REM   ******************
3000 PRINT"[CLEAR]":PRINT
3005 PRINT"[RVS]          [RVOFF] ⌐EC
     ODIFICA CODICE FOTO [RVS]
        [RVOFF]"
3010 PRINT:INPUT "⌐ODICE DELLA FOTO
     GRAFIA";CF$:PRINT"[3 DOWN]"
3019 REM     CODICE ERRATO
3020 CF=LEN(CF$):CD=CF-6:IF CF<7 TH
     EN PRINT"\\\\♠♠\⌐/⌐ ⌐—♠!♠":GO
     TO 3000
3040 FF=VAL(CF$):FD=VAL(RIGHT$(CF$,
     6)):DD=FF-FD
3060 FOR X=1 TO 6:BB$(X)=MID$(CF$,X
     +CD,1):B(X)=VAL(BB$(X))
3080 FOR J=0 TO B(X):READ D$:NEXT:P
     RINTD$
3090 IF B(X)<9 THEN FOR J=B(X)+1 TO
     9:READ D$:NEXT
3100 NEXT:RESTORE
```

```
3120 PRINT:PRINT"‾ECODIFICATA FOTO
     N."DD/1900000
3140 PRINT"♠LTRA FOTO DA DECODIFICA
     RE ? [RVS]♥[RVOFF]     [RVS]/[R
     VOFF]"
3180 GOSUB 5000
3200 IF XX$="S" THEN 3000
3220 GOTO 200
3496 REM ******************
3497 REM * STAMPA ARCHIVIO *
3498 REM ******************
3499 REM  CONTROLLARE LA STAMPANTE
3500 PRINT"[CLEAR]":PRINT:PRINT"[RV
     S]        [RVOFF] ┤Γ/I ┘┌┐└┘♠_‾ L
     ♠ ♥!♠\┐♠/I‾ [RVS]         [RVOFF
     ]":PRINT
3519 REM   SPAZIO
3520 PRINT:PRINT"    BATTERE ♥┐♠♠\Γ
     PER CONTINUARE"
3540 GOSUB 5000
3560 PRINT:PRINT"    ♥TAMPA ARCHIVIO
     DALLA FOTO N.":INPUT N:IF N=0
     THEN N=1
3580 PRINT:PRINT"    ALLA FOTO N.":I
     NPUT T
3600 OPEN 4,4
3620 PRINT#4,CHR$(14)"             ST
     AMPA ARCHIVIO"
3640 PRINT#4:PRINT#4,CHR$(15)
3650 TA=0
3660 FOR I=N TO T:TA=TA+1
3670 IF TA/2=INT(TA/2) AND LEN(A$(I
     ))<38 THEN GOSUB 3800:GOTO 374
     0
3675 IF LEN(A$(I))>38 THEN PRINT#4,
     A$(I):GOTO 3740
3680 PRINT#4,A$(I);
3720 IF A$(I)="" THEN I=T
3740 NEXT
3760 FOR XX=1 TO 5:PRINT#4,CHR$(10)
     :NEXT
3780 PRINT#4:CLOSE 4:GOTO 200
3800 PRINT#4,CHR$(16)CHR$(52)CHR$(4
     8)A$(I):RETURN
3997 REM  *********************
3998 REM  * RICERCA IN MEMORIA *
3999 REM  *********************
4000 PRINT"[CLEAR]":PRINT:PRINT"[RV
     S]        [RVOFF] ┘ICERCA IN
     MEMORIA [RVS]          [RVOFF
     ]"
4010 PRINT:PRINT"\MMETTERE I DATI C
     HE SERVONO PER LA"
4020 PRINT"RICERCA: QUANDO IL DATO
     NON SERVE"
4040 PRINT"BATTERE  - * -"
4050 PRINT:PRINT"RICHIESTA STAMPA
       [RVS]S[RVOFF]    [RVS]N[RVOF
     F]
4060 GOSUB 5000:PRINT"[CLEAR]"
4100 FOR K=0 TO 5
4120 FOR J=0 TO 9:READ D$:PRINT TAB
     (5)D$:NEXTJ
4130 INPUT B$(K):PRINT"[CLEAR]"
4135 IF B$(K)="*" THEN 4145
4140 IF B$(K)<"1" OR B$(K)>"9" THEN
      PRINT"CARATTERE ERRATO     -
     VALE *":B$(K)="*"
4145 NEXT
4150 FOR I=1 TO 2000:R=LEN(A$(I)):M
     =6:IF A$(I)="" THEN I=2000:GOT
     O 4220
4155 FOR K=0 TO 5:M=M-1
4160 IF B$(K)="*" THEN NEXTK:GOTO 4
     220
4180 IF B$(K)<>MID$(A$(I),R-M,1) TH
     EN K=0:M=0:NEXTI
4200 NEXTK
4220 R=0:RESTORE :PRINTA$(I):GOSUB
     4350:NEXTI
4260 PRINT:PRINT"ALTRA RICERCA ?
         [RVS]S[RVOFF]           [RV
     S]N[RVOFF]"
4280 GOSUB 5000
4300 IF XX$="S" THEN 4000
4320 GOTO 200
4350 IF XX$="S" THEN OPEN 4,4:PRINT
     #4,A$(I):PRINT#4:CLOSE 4
4380 RETURN
4497 REM *************
4498 REM * VARIAZIONI *
4499 REM *************
4500 PRINT"[CLEAR]":PRINT"[RVS]
             [RVOFF] XARIAZIONI [
     RVS]            [RVOFF]"
4510 PRINT:INPUT "NUMERO DELLA FOTO
      DA VARIARE";I
4520 A$(I)="":N=I:T=0
4540 GOTO 1100
4896 REM **************
4897 REM * BYTE LIBERI *
4898 REM **************
4900 BL=FRE(0)-(FRE(0)<0)*65536:RET
     URN
5000 GET XX$:IF XX$="" THEN 5000
5020 RETURN
```

Quando ho avuto l'idea di spronare Voi lettori ad inviare dei lavoretti costituiti da una sola riga, ebbene non speravo certo in un tale successo.
È proprio per questa risposta congrua ed immediata che ho deciso di aprire una rubrica apposita chiamata appunto "una riga".
Finché ci saranno idee, fintanto che Voi lettori invierete programmini con la caratteristica di questa rubrica, questa vivrà mantenendo quel clima di freschezza, di curiosità e di originalità

1 RIGA

Un programma costituito da una sola riga, per sua natura non può sortire particolari complessità. Non è detto però che tra tutte le idee di una riga non ce ne possa essere una che possa essere utilizzata all'interno di un programma complesso per migliorarne l'aspetto estetico, la potenzialità, od altro ancora.
Tutti i lettori che hanno inviato i propri elaborati hanno precisato che occorre digitare "le righe" con le abbreviazioni consentite dall'interprete BASIC del Commodore. Tutto ciò poteva essere implicito, ma è altrettanto giusto precisarlo. Per questa ragione riporto l'elenco completo delle varie parole BASIC con le relative possibili abbreviazioni (questo elenco è anche presente sul manuale nella scatola del vostro computer).

Ritengo che oramai tutti i lettori ed utilizzatori di Commodore siano già a conoscenza della possibilità di poter digitare ad esempio P seguito dalla O con lo shift ottenendo così il medesimo effetto di quando si digita POKE. Nella tabella ed in particolare nelle colonne inerenti alle abbreviazioni, tutte le lettere in maiuscolo devono essere scritte con il tasto shift premuto.
I caratteri digitali normalmente possono apparire in maiuscolo mentre appariranno nel formato equivalente al carattere grafico quelli digitati con lo shift. Ancora appariranno in minuscolo quelli digitati, normalmente, mentre saranno in maiuscolo quelli digitati con lo shift. Tutto ciò dipende dal fatto che sia stato o no premuto il tasto con il simbolo Commodore insieme allo shift.
Veniamo ora ai Vostri lavori tenendo presente che nessuno di questi ha subito qualsivoglia votazione per essere pubblicato e tenendo anche conto che in caso di uguaglianza vale purtroppo chi è arrivato prima. A presto.

Gioriano Rossi

#001

L'arlecchino. Lo schermo durante l'esecuzione viene riempito in bassa risoluzione grafica, di una sequenza ripetuta di quadrati colorati; ad ogni completamento di schermo il programma riprende con una diversa sequenza individuata casualmente.

(Roberto Biasutto)

```
1 U%=RND(U%)*20+6:FOR I=1024 TO 2
  023 STEP U%:FOR J=0 TO U%:POKE
  I+J,160:POKE 54272+I+J,J:NEXTJ,
  I:GOTO 1
```

#002

Indovina indovinello. Questa "una riga" contiene un semplice giochino che permette di indovinare una lettera che viene scelta dal Commodore a caso.

(Nicola Montanari)

```
1 A=RND(1)*26+65:INPUT "LETTERA";
  B$ IF B$<>CHR$(A) THEN PRINT"NO
  ":GOTO 1
```

#003

Caratteri a caso. Ecco un programma che è in grado di mostrare, casualmente, sullo schermo tutti i caratteri del Commodore.

(José Andreani)

```
1 A=1024:B=2023:X=RND(1)*256:FOR
  K=A TO B:POKE K,X:NEXT:Y=RND(1)
  *X:FOR K=B TO A STEP -1:POKE K,
  Y:NEXT: RUN
```

#004

Multipli. Il programma stampa in continuazione i multipli (fino a cento) di una qualsiasi cifra introdotta da tastiera. È ovvio che all'inizio saranno visualizzati una serie di zeri fintanto che non verrà battuta una cifra e continuerà con questa fino all'introduzione di una cifra successiva. Ogni altra spiegazione è naturalmente superflua.

(Dino Ticli)

```
1 GET A$:N=VAL(A$):PRINT"[VERDE][
  RVS]MULT.DI"N:FOR T=1 TO 100:PR
  INT"[ROSSO]▒[BLEU]"N*T,:FOR I=1
   TO 100:NEXTI,T:GOTO 1
```

#005

Titoli in movimento. La riga consente, mediante l'uso della funzione COS (X) presente nell'interprete del Commodore, di osservare lo scorrere sullo schermo di una qualsiasi stringa di caratteri, non più lunga di 12 caratteri. Lo scorrere avverrà in maniera sinusoidale.

(Nicoletta Berliri)

```
1 FOR S=0 TO 6.4 STEP .3:PRINT TA
  B(10*COS(S)+16);"TITOLATRICE":F
  OR T=0 TO 10:NEXTT:NEXTS:GOTO 1
```

#006

Di tutti i colori. Dopo aver digitato il programmino eseguite il RUN e tutto lo schermo diventerà di un solo colore. Premendo i tasti funzione 1 o 3 si otterrà il cambiamento del colore.

(Massimiliano Aldieri)

```
1 A=53280:GET A$:B=A$="[TF1]":D=A
  $="[TF3]":C=C-B:E=E-D:POKE A,C
  AND 15:POKE A+1,E AND 15:GOTO 1
```

#007

Ancora colori. Il programma si limita a mostrare la gamma dei colori prevista per il Commodore 64 e sfrutta le consuete POKature agli indirizzi 53280 e 53281.
(Questa 1R si differenzia dalla precedente solamente per l'automatismo del cambiamento colore senza sminuire o aumentare il valore dell'operato di Massimiliano o di Stefano. NDR)

(Stefano Nocilli)

```
1 FOR A=0 TO 15:FOR B=0 TO 15:POK
  E 53280,A:POKE 53281,B:FOR H=1
  TO 1000:NEXTH:NEXTB:NEXTA
```

#008

N fattoriale. perché scrivere programmi lunghi per eseguire dei calcoli? Ecco qui un programma per il calcolo di N fattoriale. (per non incorrere in errori N non potrà essere superiore al numero 33. NDR)

(Giacomo Salmeri)

```
1 INPUT N:K=1:FOR I=1 TO N:K=K*I:
  NEXT:PRINTK:GOTO 1
```

#009

L'uccellino impazzito. Un pò di movimento sullo schermo non guasta mai. Ecco allora un uccellino che "vola" sul video. (ho voluto inserire una piccola modifica alla riga di Livio in maniera tale da dare maggior effetto al risultato. NDR)

(Livio Galante)

```
PRG ORIGINALE

1 A$=" ╱●╲":B$=" ╲●╱":I=INT(RND(1)*
  255)+1:PRINT"[CLEAR]":PRINTSPC(
  I)A$:PRINT"[CLEAR]":PRINTSPC(I)
  B$:GOTO 1
```

```
PRG MODIFICATO

1 A$=" ╱●╲":B$="[3 LEFT]╲●╱":I=INT
  (RND(1)*255)+1:PRINT"[CLEAR]"SP
  C(I)A$;:FOR A=1 TO 200:NEXT:PRI
  NTB$:GOTO 1
```

#010

Tutto è casuale. In una prima fase dell'esecuzione potrebbe sembrare banale, ma dopo qualche instante diventa simpatico. Provare per credere! Il programma si basa sulla mappa di memoria di schermo e quella di colore. Tutto è casuale: posizione, carattere e colore.

(Bruno Giacosa)

```
1 T=RND(1):POKE (T*1000+55296),RN
  D(1)*16:POKE (T*1000+1024),RND(
  1)*254+1:GOTO 1
```

#011

Ancora a caso. Invece del carattere casuale, ovvero l'RND (1) *255, può essere sostituito ad esempio da uno fisso come ad esempio l'81.

(Bruno Giacosa)

```
1 PRINTCHR$(RND(1)*255);:FOR I=1
  TO 5:NEXT:GOTO 1
```

#012

A dollaro. Dopo aver personalizzato la variabile A$ in maniera diretta eseguire il GOTO 1 e non RUN. Il contenuto della variabile verrà visualizzato in modo interessante.

(Daniele Mazzucca)

```
1 FOR N=1 TO LEN(A$):PRINT"[HOME]
  [DOWN][LEFT]"MID$(A$,N,1);"[HOM
  E] "CHR$(20);:FOR M=0 TO 99:NEX
  TM,N:GOTO 1
```

#013

Uno o due. Questa "una riga" può essere scritta abbreviando solamente l'istruzione PRINT con il classico "?". Dopo il RUN si potrà giocare. Il computer "pensa" un numero che potrà essere solamente uno oppure due. Successivamente noi dobbiamo indovinare il numero. In base alla risposta il computer totalizzerà la vincita. (tengo a evidenziare la soluzione che Daniele ha trovato affinché la totalizzazione sia eseguita senza la necessità di utilizzare una IF che avrebbe compromesso l'obbligatorietà dell'unicità della riga. NDR)

(Daniele Mazzucca)

```
1 PRINT"HAI $"S:A=INT(RND(1)*2+1)
  :INPUT "CHE NUMERO";B:S=S+(0-AB
  S(A-B)*2+1):PRINTA:GOTO 1
```

#014

Rimbalzi. Con un pizzico di fantasia si possono osservare una serie di palline che rimbalzano da un lato all'altro dello schermo. (ho voluto suggerire una ulteriore "una riga" per poter evidenziare il rimbalzo di una pallina o asterisco da una parte all'altra dello schermo. NDR)

(Marco Busnelli)

```
PRG ORIGINALE

1 FOR I=0 TO 39:PRINT TAB(I)"*":N
  EXT:FOR I=39 TO 0 STEP -1:PRINT
   TAB(I)"*":NEXT:GOTO 1

PRG MODIFICATA

1 FOR I=1 TO 39:PRINT"[LEFT] *";:
  FOR A=1 TO 99:NEXTA,I:FOR I=39
  TO 1 STEP -1:PRINT"* [3 LEFT]";
  :FOR A=1 TO 99:NEXTA,I:GOTO 1
```

#015

Viva l'Italia. Il programma disegna, a tutto schermo, la bandiera italiana. Il suo funzionamento è estremamente semplice. Con i cicli FOR... NEXT si stampa uno spazio in reverse-mode in modo consecutivo per 14 volte per il colore rosso e 13 volte per i colori bianco e verde. (l'idea mi è piaciuta ed ho voluto vedere se era possibile vedere il medesimo effetto in altri modi. Eccone due. NDR)

(Guido Baldini)

```
PRG ORIG

1 FOR K=1 TO 14:PRINT"[RVS][ROSSO
  ] [RVOFF]";:NEXT:FOR K=1 TO 13:
  PRINT"[RVS][BIANCO] [RVOFF]";:N
  EXT:FOR K=1 TO 13:PRINT"[RVS][V
  ERDE] [RVOFF]";:NEXT:GOTO 1

PRG MODIFICA 1

1 PRINT"[RVS][ROSSO]
   "SPC(13)"[VERDE]              ":
  GOTO 1

PRG MODIFICA 2

1 A$="              ":PRINT"[RVS][
  ROSSO]"A$SPC(14)"[VERDE]"A$;:GO
  TO 1
```

#016

I mattoni. Come si può riempire lo schermo di mattoni in modo tale che sembri un muro senza fine? Ecco "una riga" che aiuta allo scopo. I simboli grafici si ottengono; tasto Commodore e P, tasto shift e L e quindi ancora tasto Commodore e P.

(Raffaele Silvestri)

```
1 PRINT"[ROSSO][UP]_L_[DOWN]";:FO
  R K=1 TO 30:NEXT:GOTO 1
```

#017

Il conta Tempo. Il vostro Commodore ha un orologio preciso. Sapete che è possibile personalizzarlo ed utilizzarlo? Le variabili di sistema sono TI e TI$, abbreviazioni di TIME e TIME$. In TI si ha il tempo espresso in sessantesimi, mentre in TI$ il tempo è espresso in ore, minuti e secondi.

Il concetto di base parte dall'istruzione PRINT TI che visualizza il conteggio dell'orologio interno del Commodore. Assegnando alla variabile S il valore di TI diviso 60 si ha il conteggio in secondi e relativi decimi, centesimi e così via. L'incremento del clock avviene automaticamente tramite una delle azioni periodiche del microprocessore. Per ottenere le ore ed i minuti interi è stata usata l'istruzione BASIC INT, la quale è stata impiegata per definire i valori delle variabile H per le ore e M per i minuti. Ad H viene assegnato il valore calcolato sulla variabile S fratto 3600 corrispondente ai secondi di un'ora, mentre ad M viene assegnato il valore corrispondente dall'espressione di S fratto 60 e per evitare che M sia uguale al totale dei minuti trascorsi oltre l'ora viene sottratto 60*H. Le istruzioni PRINT presenti nella seconda parte di questa "una riga" visualizzano le tre variabili interessate al conteggio:

COMPUTER QUESTO MESE È QUESTO

COMPUTER

N. 72 - lire 3500 il "NEWSMAGAZINE" dell'informatica

Buongiorno, robot

Memorie di massa quali scegliere

Cristalli liquidi alla riscossa

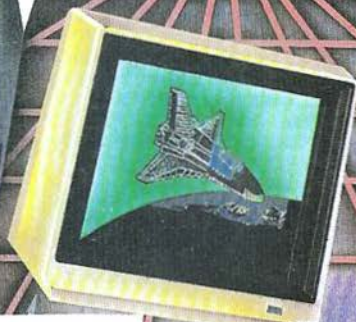
Quinta generazione la sfida giapponese

Benchmark ICL modello 16

I packages grafici per il manager

Ariston Francesco '84

systems

Computer - Anno VIII - n. 72 - 25 dicembre 1984 - 25 gennaio 1985 - Sped. Abb. Post. Gr. III/70 - Distr. MePe

S, M e H.
La personalizzazione del clock può avvenire se necessario in maniera diretta semplicemente con: TI$="122530" se nel momento in cui si premerà il tasto RETURN sarà mezzogiorno e venticinque minuti e trenta secondi. Fatto ciò si può impartire il comando RUN.

(Gabriele Mercati)

```
1 S=TI/60:H=INT(S/3600):M=INT((S/
  60)-(60*H)):PRINT"[HOME]H"H:PRI
  NT"M"M:PRINT"[RVS]"S: RUN
```

#018

Di tutti i colori. La locazione di memoria 162 serve per poter variare il colore del cursore, contiene un contatore ciclico che va da 0 a 255. Il Commodore, sia esso 64 o VIC 20, utilizza questo contatore per aggiornare la variabile temporale TI$.

(Eugenio Pollari)

```
1 PRINT"[RVS]     ";:POKE 646,PEEK(
  162):GOTO 1
```

#019

Hardcopy. Questa "una riga" permette di ottenere l'hardcopy dello schermo sulla stampante con numero di device corrispondente a 4. Basta cambiare opportunamente il numero di device ed ecco possibile l'hardcopy anche sulla stampante plotter 1520.

(Marco Melloni)

```
1 OPEN 3,3:OPEN 4,4:PRINT"[HOME]"
  ;:FOR I=0 TO 999:GET #3,A$:PRIN
  T#4,A$;:NEXT:CLOSE 3:CLOSE 4
```

#020

Giriamo il video. Marco, l'autore della precedente "una riga", ha inviato anche una seconda realizzazione, che però aveva una leggera imperfezione. Ho ritenuto comunque che l'idea fosse ottima ed ho così modificato l'elaborato: il video viene aperto in input come se fosse in file sequenziale ed in un secondo tempo ritorna alla normalità di output.

(da una idea di *Marco Melloni*)

```
1 OPEN 3,3:PRINT"[HOME]";:INPUT#3
  ,A$:PRINT"[24 DOWN]"A$;:CLOSE 3
  :GOTO 1
```

#021

Un altro orologio. Sfruttando ancora la variabile temporale TI$ Silvano ha realizzato un orologio digitale a colori. A differenza #017 abbiamo qui una manipolazione della variabile TI$ piuttosto che la variabile TI.

(Silvano Fulchero)

```
1 PRINT"[CLEAR][VERDE]";"ORE[NERO
  ]";LEFT$(TI$,2);"[VERDE] MIN.[N
  ERO]";MID$(TI$,3,2);"[VERDE]SEC
  .[NERO]";RIGHT$(TI$,2):GOTO 1
```

#022

PRINT AT. Sfruttando una routine del Kernal simuliamo l'istruzione AT non presente nel BASIC del Commodore. La presente "una riga" è quindi un semplice dimostrativo della possibilità di utilizzo delle istruzioni in programmi più complessi.

(Salvatore Daidone)

```
1 PRINT"[CLEAR]":FOR A=1 TO 23:X=
  A:Y=24-A:POKE 211,X:POKE 214,Y:
  SYS58732:PRINT"[RVS]-AT-[RVOFF]
   PER CBM 64":NEXT:GOTO 1
```

#023

CHR$ e POKE. L'attuale 1R consente di analizzare la corrispondenza fra il codice in CHR$ ed il codice in POKE. IL funzionamento è alquanto semplice: dai caratteri provenienti dall'INPUT vengono visualizzati i corrispondenti CHR$ e codici numerici per i POKE.

(Massimo Dossi)

```
1 PRINT"[CLEAR]":INPUT N$:FOR L=1
   TO LEN(N$):POKE 1145+L,ASC(MID
  $(N$,L,1)):NEXT:PRINT"[DOWN]"
```

#024

Geometria. Con questa semplice 1R si può calcolare una qualsiasi area di un quadrato o di un rettangolo.

(Luigi Dallanoce)

```
1 INPUT "[CLEAR]BASE";B:INPUT "[D
  OWN]ALT.";A:C=B*A:PRINT"AREA="C
  :FOR D=1 TO 3000:NEXT:GOTO 1
```

#025

Dal lordo al netto. Con una riga è possibile calcolare l'importo netto proveniente da uno lordo di cui si conosce l'aliquota.

(Giuseppe Dallanoce)

```
1 PRINT"[CLEAR]":INPUT "TOTALE  "
  ;T:INPUT "ALIQUOTA";A:N=T-(T*A/
  (100+A)):PRINT"[HOME][3 DOWN]
  NETTO[3 RIGHT]=";N
```

#026

Toto random. Questa 1R è destinata alla generazione casuale di 13 simboli come nell'uso delle norme del totocalcio. La colonna di numeri viene evidenziata sul video affiancata dal numero della giocata corrispondente. Modificando il valore posto nell'istruzione IF si può cambiare a piacimento il numero delle giocate.

(Gianfranco Dallanoce)

```
1 A$="12X":B$=MID$(A$,RND(1)*3+1,
  1):A=A+1:PRINTB$;A:IF A<13 THEN
   1
```

#027 e #028

Il verme viaggiatore e pazzi saluti. Ecco due versioni della medesima possibilità di manipolare una stringa di caratteri.

(Daniele Novis)

```
1 D$="[LEFT]":FOR X=1 TO 200:PRIN
  T"[CLEAR]":PRINTD$;"O●O●O●O●O";
  :D$=D$+"[RIGHT]":NEXT

1 A$="[RIGHT]":FOR I=1 TO 200:PRI
  NT"[CLEAR]":FOR A=1 TO 80:PRINT
  A$;"CIAO!";:A$=A$+"[RIGHT]":NEX
  TA:NEXTI
```

#029

Tutti sul video. Volete vedere byte per byte i contenuti di tutte le locazioni di memoria? Ebbene battete questa semplice 1R.

(Gianfranco Basso)

```
1 PRINTCHR$(PEEK(N));:N=N+1:GOTO
  1
```

#030

Caleidoscopio. Questo programma va a scrivere sulla mappa video una serie casuale di caratteri e così pure casualmente anche i colori relativi.

(Michele Giuri)

```
1 PRINT"[CLEAR]":FOR I=1024 TO 20
  23:POKE I,RND(1)*191:POKE I+542
  72,RND(1)*15:NEXT:GOTO 1
```

#031

Dove Commodore? Lo schermo si riempirà delle lettere comprese nella parola Commodore. Le posizioni sono casuali. Sarà divertente individuare dove la casualità scriverà esattamene il nome del Vostro computer.

(Maurizio Brandimarti)

```
1 FOR I=1 TO 6:READ A$(I):NEXT:X=
  INT(RND(1)*6)+1:PRINT"[RIGHT]"A
  $(X);:DATA C,O,M,O,R,E:RESTORE
  :GOTO 1
```

ABS	A SHIFT B	A	LEFT$	LE SHIFT F	LE	SGN	S SHIFT G	S
AND	A SHIFT N	A	LEN	NONE	LEN	SIN	S SHIFT I	S
ASC	A SHIFT S	A ♥	LET	L SHIFT E	L	SPC(	S SHIFT P	S
ATN	A SHIFT T	A	LIST	L SHIFT I	L	SQR	S SHIFT Q	S ●
CHR$	C SHIFT H	C	LOAD	L SHIFT O	L	STATUS	ST	ST
CLOSE	CL SHIFT O	CL	LOG	NONE	LOG	STEP	ST SHIFT E	ST
CLR	C SHIFT L	C	MID$	M SHIFT I	M	STOP	S SHIFT T	S
CMD	C SHIFT M	C	NEW	NONE	NEW	STR$	ST SHIFT R	ST
CONT	C SHIFT O	C	NEXT	N SHIFT E	N	SYS	S SHIFT Y	S
COS	NONE	COS	NOT	N SHIFT O	N	TAB(	T SHIFT A	T ♠
DATA	D SHIFT A	D ♠	ON	NONE	ON	TAN	NONE	TAN
DEF	D SHIFT E	D	OPEN	O SHIFT P	O	THEN	T SHIFT H	T
DIM	D SHIFT I	D	OR	NONE	OR	TIME	TI	TI
END	E SHIFT N	E	PEEK	P SHIFT E	P	TIME$	TI$	TI$
EXP	E SHIFT X	E ♠	POKE	P SHIFT O	P	USR	U SHIFT S	U ♥
FN	NONE	FN	POS	NONE	POS	VAL	V SHIFT A	V ♠
FOR	F SHIFT O	F	PRINT	?	?	VERIFY	V SHIFT E	V
FRE	F SHIFT R	F	PRINT#	P SHIFT R	P	WAIT	W SHIFT A	W ♠
GET	G SHIFT E	G	READ	R SHIFT E	R			
GET#	NONE	GET#	REM	NONE	REM			
GOSUB	GO SHIFT S	GO ♥	RESTORE	RE SHIFT S	RE ♥			
GOTO	G SHIFT O	G	RETURN	RE SHIFT T	RE			
IF	NONE	IF	RIGHT$	R SHIFT I	R			
INPUT	NONE	INPUT	RND	R SHIFT N	R			
INPUT#	I SHIFT N	I	RUN	R SHIFT U	R			
INT	NONE	INT	SAVE	S SHIFT A	S ♠			

Tabella abbreviazione delle parole chiave del Basic.

Scambiatevi le liste

Tutte le richieste di Vendo-Scambio-Compro programmi per... verranno citate esclusivamente con il solo nome, cognome ed indirizzo. Sta a voi lettori scambiarvi le varie liste. Verranno pubblicati integralmente a giudizio del caporedattore, quegli annunci che non fanno parte della categoria di cui sopra. Buoni scambi e vendite!!

(Gloriano Rossi)

Bifolchi Giordano - Via nel Corso 111 - 53045 Montepulciano (SI) - Tel. 0578/757907-716397.

Lino Italia - L.go C. Volpicelli 6/A - 80136 Napoli - Tel. 081/348082.

Gaetano Di Mattia - C.so Vittorio Emanuele 224 - 93012 Gela (CL) - Tel. 0933/917565.

Ropele Andrea - Via Marconi 30 - 39065 Laives (BZ) - Tel. 0471/951480.

Salmaso Massimiliano - Via Voltan 5 - 30039 Stra (VE).

Maio Antonio - Via S. Agostino 9 - 98057 Milazzo (MI) - Tel. 090/921261.

Marcello Cesi - Via Migliana Nuova 178 - 00146 Roma - Tel. 06/5266009 dopo le ore 20.

Augusto Bernardini - Via Valle Verde 5 - 05100 Terni - Tel. 0744/56870-47148.

Quaglia Roberto - Via Martinazzoli 2 - 20161 Milano - Tel. 02/6462130 dalle ore 21 alle 22.

Salvati Massimo - Via Piero Colonna 30 - 00149 Roma - Tel. 06/5261016 ore 10/12.30 oppure 17/19.30.

Stefano Massoli - Via Massari 10 - 06100 Perugia - Tel. 28983.

Natali Sandro - Via XIV Luglio 36/A - 50019 Sesto Fiorentino (FI) - Tel. 055/4481640 ore pasti tutti i giorni.

Lucio Tonucci - Viale Romagna 36 - 61032 Fano - Tel. 879424 ore pasti (13.30 oppure 20).

Arturo Viscardi - Via Brosetta 16 - 24100 Bergamo - Tel. 035/239150 dalle 18 in poi.

Andrea Toschi - L.go Garibaldi 32 - 41100 Modena.

Manzo Fabio - Via Calabria 36 - 81022 Casagiove (CE) - Tel. 0823/466177 dalle 14 alle 16 dalle 21 alle 23.

Club 64 Commodore casella postale 6 - 35044 Montagnana (PD) - Tel. 0429/82469 dalle 20 in poi.

Carrone Vincenzo - Via G. Pascoli 67 - 86100 Campobasso - Tel. 0874/91995 ore pasti.

Maurizio Caporale - Viale della Rimembranza 29 - 66034 Canciano (CM) - Tel. 0872/27296 ore pasti.

Roberto Cotza - Via Puccini 60 - 20099 Sesto San Giovanni (MI) - Tel. 02/2425392 dalle ore 12.30 alle 15.30.

Mariano Talamo - Via Colucci 10 - 70019 Triggiano (BA) - Tel. 080/681470 dopo le ore 20.

Elia De Falco - Via Roma 14 - 84036 Sala Consilina (SA) - Tel. 0975/21193 dalle ore 22 alle 23.

Claudio Veronese - Via Tonso 15 - 15100 Alessandria.

Grossi Alessandro - Via Dario Campana 19 - 47037 Rimini - Tel. 0541/773458 dalle 16 alle 20.

Salvatore Raucci - Via G. D'Annunzio 1 - 81100 Caserta - Tel. 0823/323745 ore pasti.

Germano Fasano - Via Ercole I D'Este 3 - 44100 Ferrara - Tel. 0532/25401 dalle ore 13.30 alle 14.30.

Alessandro Sofia - Via Cremona 42 - 63040 Folignano (AP) - Tel. 0736/68148 dalle 13.30 alle 14 oppure dalle 20 alle 21.30.

Carmine Parrella - Via T. Mommsen 2 - 82100 Benevento - Tel. 0824/48068 dalle 14 alle 17.

Quargnali Marco - Via Nazionale 73 - 33048 S. Giovanni Al Nat. (UD).

Galanti Marcello - Via Torre Nuova 134 - 00133 Roma.

Ferrari Fabio - Via Araldi 5 - 29100 Piacenza - Tel. 0523/753283 dopo le ore 18.

Cristiano Garibaldi - Via Vicolo S. Lazzaro 16 - 18038 Sanremo (IM) - Tel. 0184/823554 ore pasti (12/14 oppure 19/21).

Frazzano Antonio - Via De Nicola 10 - 21036 Lucera (FG).

Gianni Gaudino - Via Graglia 18 - 10136 Torino - Tel. 011/352830 dalle ore 19.30 alle 21.30.

Stefano Sarcletti - Via Doss Di Pez 17 - 38023 Cles (TN) - Tel. 0463/21930 solo ore serali.

Nicola Narni - Via Filippo Masci 40 - 86100 Chieti

Giovanelli Claudio - Via Ripamonti 194 - 20141 Milano - Tel. 02/536926.

Alfaro Ceccarini - Via Di Vittorio 10 - 58028 Follonica - Tel. 0566/43248 dopo le ore 19 nei giorni feriali

Andrea Rigoni - Via G. Verdi 16 - 35031 Abano Terme - Tel. 049/811447 ore pasti.

Jory Fausto - Viale De Gasperi 49 - 38023 Cles (TN) - Tel. 0463/21660 dopo le 20.

Lai Piergiuliano - Via G. Zuddas 143 - 09100 Cagliari - Tel. 562668 dalle 13 alle 17.30.

Cravangola Cristiano - Via Prof. Oliva 16 - 12011 Borgo S. Dalmazzo (CN) - Tel. 0171/

769659 pomeriggio o sera.

Alfredo Dal Ferro - Via Fornace 3 - 21040 Castronno - Tel. 0332/495989 dopo le 19.

Vitagliano Fabrizio - Via Servais 73 - 10146 Torino - Tel. 011/720656 serali dopo le 20.

Master Computer Club - C.so Piave 20 - 06023 Gualdo Tadino (PG) - Tel. 075/913234 ore pasti.

Marcorin Gian Alberto - Via Falconera 45 - 30021 Caorle (VE) - Tel. 0431/819662 dalle 14 alle 15.

Angelo Orlandi - Via Delle Albizzie 4 - 00152 Roma - Tel. 06/288368 dalle 13 alle 21.30.

Raineri Antonio - Via Mazzini 1 - 20077 Melegnano (MI) - Tel. 02/4833788.

Cotogni Gianni - Via Strambino 23 - 10010 Carrone (TO) - Tel. 0125/712311.

Renato Menegatti - Via Piemonte 26 - 20052 Monza (MI).

Roberto Bianchi - Via Ardigò 1 - 20052 Monza (MI) - Tel. 039/360493.

Sandro Natali - Via XIV Luglio 36/A - 50019 Sesto Fiorentino - Tel. 055/4481640.

Bombara Gianpaolo - Via Graglia 36 - 10136 Torino - Tel. 011/363547.

Roberto Franzoso - P.zza Medaglie d'Oro 8 - 14100 Asti.

Fabio Invernizzi - Via Montecervino 13 - 20066 Melzo (MI) - Tel. 02/95711176.

Michele Pozzi - Via Coll. Maria Luigia 15 - 43100 Parma - Tel. 0521/32165.

Carobene Giovanni - P.zza Dante 89 - 80135 Napoli.

Bompieri Silvano - Via Boccaglioni 8 - 46040 Monzambano - Tel. 0376/845372.

Lena Gregorio - V.le Silvani 3/2 - 40122 Bologna - Tel. 051/551178.

Stramigioli Maurizio - Via G. Pascoli 67 - 47033 Cattolica (FO).

Dino Marrocchi - Via Marconi 302 - 65100 Pescara - Tel. 085/68352.

Luca Dell'Anna - Via Avellino 12 - 73100 Lecce - Tel. 0832/591157.

Fabio Molinari - Via Gerolamo Belloni 112 - 00191 Roma.

Marino Russo - Via Sandro Gallo 169 - 30126 Venezia Lido - Tel. 041/30777-768593.

Bellofiore Giuseppe - Via Aurelia 502 - 57012 Castiglioncello (LI) - Tel. 0586/752135.

Claudio Veronese - Via M. Bensi 29 - 15100 Alessandria - Tel. 0131/344591.

Cellesca Giuseppina - P.zza Papa Giovanni XXIII 6 - 01010 Blera (VT) - Tel. 06/9027228.

Canè Cirio - Via Arcora 7 - 80013 Casalnuovo - Tel. 8421488.

Vicario Cesare - Via Pr. Umberto 44 - 28021 Borgomanero - Tel. 0322/82509.

Sani Massimiliano - Via S. Rosa 2 - 56048 Volterra (PI) - Tel. 0588/88987.

Stefano Rossi - Via Oratorio 43 - 52011 Bibbiena (AR) - Tel. 0575/593854.

Luigi Palumbo - V.le A. Gramsci 26 - 80122 Napoli - Tel. 081/660068.

Ubaldo Mancini - Via Valdellatorre 174 - 10151 Torino - Tel. 011/7391285.

Massimiliano Gregnanin - Via XXV Aprile 4 - 10036 Settimo Torinese - Tel. 011/8010392.

Marco di Giorgio - V.le Alessandrino 251 - 001172 Romma - Tel. 06/2819659.

Giuseppe Genchi - Via Ummberto Maddalena 1/2 - 90132 Palermo - Tel. 590141.

Paolo Zancarlin - Via Cannareggio 2978/A - 30121 Venezia - Tel. 041/715422.

Fabrizio Rizzi - Via Castello 3060/B - 30122 Venezia - Tel. 041/22883.

Carmelo Briguglio - Via Villa Franca 19 - 98026 Nizza Di Sicilia (ME).

Massimo Proia - Via Pubblico Passeggui 19 - 29100 Piacenza - Tel. 0523/32417.

Roberto Chimento - Via Luigi Rizzo 18 - 80124 Napoli - Tel. 081/617368.

Arnaldo Restrelli - Via Melzi 135 - 20025 Legnano (MI) - Tel. 0331/596582.

Domus - Via Crimea 3 - 20100 Milano - Tel. 02/4049179-5696868.

Eligio Dati - Via Marignana 118 - 55041 Canaiore - Tel. 0584/980970.

Giovanni Addabbo - Via Monte Vettore 46 - 60131 Ancona.

Mauro Straulino - Via Colli Portuensi 242 - 00151 Roma - Tel. 06/5343658.

Pagliari Alessandro - Via Ferrata 9 - 61011 Gabice - Tel. 0541/967391.

Giuseppe - Via Mameli 15 - 33100 Udine - Tel. 0432/291665.

Ferraguti Francesco - Via Mazzini 1348 - 41043 Formigine (MO) - Tel. 059/570110.

Savoia Massimo - Via S. Anna 11 - 20045 Besana Brianza (MI) - Tel. 0362/940245.

Giorgio Primiceri - Via Sauro 87 - 73100 Lecce - Tel. 45344.

Ernesto Chianetta - Via Nemorense 39 - 00199 Roma - Tel. 06/8453884.

Cantalipi Stefani - Via M. Guzzi 13 - 66100 Chieti - Tel. 0871/2020.

Domenico Gallè - Via Contrada Guido 36 -

88029 Serra S. Bruno - Tel. 0963/71210.

Mario Guarneri - Via Gramsci 64 - 95018 Riposto (CT) - Tel. 932106.

Maurizio Sterneri - Via Gobetti 10 - 41012 Carpi (MO) - Tel. 059/683684.

Bruno Marzio - Via Ricordi 11 - 20131 Milano - Tel. 02/203973.

Allievi Alessandro - Via Monte Bianco 22 - 20021 Bollate (MI) - Tel. 3506085.

Ponte Gabriele - Via Borghese 1 - 20154 Milano - Tel. 02/344891.

Nadia Bricchi Camagna - Via Vinlaglio 151 - 15100 Alessandria - Tel. 0131/446521.

Adorno Antonio - Via Stazione 28 - 98060 Oliveri (RE) - Tel. 0941/33133.

Fox Software c/o Bompieri Silvano - Via Boccaglione 8 - 46040 Monzambano (MN) - Tel. 0376/845372-845216.

Claudio P. - P.zza Spotorno 3 - 20145 Milano - Tel. 02/680482.

Ezio Bovio - Via L. Miglio 32 - 28043 Bellinzago - Tel. 0321/98430.

Federico Chieli - Via Volturno 1 - 10133 Torino - Tel. 011/6506228.

Sergio Poli - Via G. Modena 35 - 35100 Padova - Tel. 049/850026.

Lorenzo Giannoni - Via Zanelli 19 - 25100 Brescia - Tel. 030/294505.

Marcelli Luca - Via Ugo Bassi 70 - 58900 Grosseto - Tel. 0564/29590.

Diego Palazzi - Via Monte Rosa 15 - 20140 Milano - Tel. 02/465802.

Marcello Caputo - Via Filangeri 1 - 71100 Foggia - Tel. 25072.

Walter Cacioni - Via Dei Mille 26 - 00185 Roma - Tel. 06/4956604.

Franco Piccinno - Viale L. Da Vinci 146 - 00145 Roma - Tel. 06/5132153.

Alberto Caciari - Via S. Allende 15 - 40139 Bologna - Tel. 051/491185.

Reale Nicola - Via Circonvallazione 118 - 95041 Caltagirone (CT) - Tel. 0933/22766.

Se vuoi abbonarti

Registrate il mio abbonamento annuale a Commodore.

☐ Ho versato oggi stesso il canone di Lire 25.000 a mezzo c/c postale nº 31532203 intestato a: Commodore Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

☐ Accludo assegno per lire 25.000 banca .... nº .... a favore di Commodore Systems Editoriale

Il mio computer è: VIC 20 ☐, C 64 ☐, altro (specificare) ....

Ho ☐ / non ho ☐ la stampante, ma voglio ☐ comprarla.

Preferisco programmi di gioco ☐, didattici ☐, d'utilità ☐, altro ....

Nome .... Cognome ....

Via .... nº .... CAP. ☐☐☐☐☐ Città ....

Tel. ....

Se vuoi collaborare

Registratemi fra i collaboratori regolari di Commodore.

A titolo di prova vi invio un articolo e la cassetta col programma "........" di cui vi garantisco l'assoluta originalità autorizzandovene la pubblicazione.

☐ Scrivetemi all'indirizzo sottoindicato ....

Nome ....

Via .... Nº ....

Tel. .... CAP .... Città ....

Se vuoi un consiglio o consigliarci

HELP ....

Nome ....

Via .... nº .... CAP. ☐☐☐☐☐ Città ....

Tel. .... Orario ....

Il mio computer è configurato:

			Sono in possesso	No	Ho intenzione di acquistare
Vic 20 ☐	espanso a ....K		☐	☐	☐
C 64 ☐			☐	☐	☐
Floppy ☐	quale: 1541 ☐	altro	☐	☐	☐
Stampante ☐	quale: MPS801 ☐	altro	☐	☐	☐
Plotter ☐	quale: 1520 ☐	altro	☐	☐	☐
Registratore ☐	quale: 1530 ☐	altro	☐	☐	☐

Televisore ☐, TV-Monitor ☐, Monitor ☐, Colore ☐, B/N ☐

Nome .... Cognome ....

Via .... nº .... CAP. ☐☐☐☐☐ Città ....

Tel. ....

Se vuoi vendere o comprare

Vendo ☐ Compro ☐ ....

Nome ....

Via .... nº .... CAP. ☐☐☐☐☐ Città ....

Tel. .... Orario ....

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio abbonarmi

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio collaborare

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, chiedo consiglio

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio votare

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si vendo/ compro

Un'iniziativa condotta con la nota rivista Computer

PROGRAMMO IN BASIC

Il linguaggio del futuro in un manuale rapido e completo di Clizio Merli
pagg. 224 (L. 9.000)

Il Basic, attualmente il linguaggio più conosciuto - adatto all'utilizzo su qualunque tipo di macchina e in particolare sul personal e gli home-computer - può essere appreso in poche ore con l'ausilio di questo agile manuale.

COME SCEGLIERE UN COMPUTER

Guida pratica per l'acquisto di un mini o di un micro computer professionale di Michele Di Pisa
pagg. 160 (L. 6.000)

Quale modello scegliere tra gli oltre 600 computer commercializzati in Italia? La conoscenza delle caratteristiche delle varie macchine è indispensabile. Con un approccio a "menu" l'Autore vuol essere guida proprio in questa fase.

UTILITY E ROUTINE PER IL COMMODORE 64

di Gloriano Rossi
pagg. 192 (L. 9.000)

L'esecuzione di una istruzione BASIC può richiedere diverse centinaia di passi di programmi in linguaggio macchina. La dimensione dei programmi è ciò che intimidisce maggiormente l'utilizzatore medio di Commodore: aiutato da questo testo chiunque potrà affrontare senza problemi il processo di scrittura di un programma.

BASIC PER LO SPECTRUM

di Maurizio Ariena e Clizio Merli
pagg. 192 (L. 9.000)

Un libro per quanti hanno acquistato il computer ZX Spectrum della Sinclair e intendono sfruttarne appieno tutte le capacità, dall'hardware alla programmazione in assembly (linguaggio macchina).

I volumi, che sono comunque in vendita nelle migliori librerie di tutta Italia, possono anche essere richiesti direttamente all'Editore.
Importante: l'ordine minimo dovrà essere di L. 15.000.

DIZIONARIO DELL'INFORMATICA

Vocabolario Inglese-Italiano di Cultrera, Di Pisa, Giacomelli
pagg. 388 (L. 25.000)

Uno strumento indispensabile per chi si avvicina al mondo dell'informatica e per gli specialisti che hanno l'esigenza di accedere alla dinamica letteratura anglosassone.

Edizioni ACANTHUS

VIALE GRAN SASSO, 23 - 20131 MILANO

Inviatemi i seguenti volumi:

Titolo	quantità	prezzo unitario
spese postali		L. 2.000
	totale	L.

Pagherò contrassegno il dovuto (più L. 2.000 per contributo spese postali) al ricevimento. Potrò restituire i libri entro 8 giorni se non saranno di mio gradimento e avere il rimborso immediato.

COGNOME

NOME

VIA N.

C.A.P. CITTÀ

FIRMA

DATA

Scrivere in stampatello e spedire in busta chiusa.